(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
3 May 2001 (03.05.2001)

PCT

(10) International Publication Number
WO 01/31479 A1

(51) International Patent Classification[7]: G06F 17/00, 17/21, 17/30

(21) International Application Number: PCT/IL00/00689

(22) International Filing Date: 26 October 2000 (26.10.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/161,712	27 October 1999 (27.10.1999)	US
60/171,586	23 December 1999 (23.12.1999)	US
60/191,122	22 March 2000 (22.03.2000)	US
09/568,988	11 May 2000 (11.05.2000)	US

(71) Applicant *(for all designated States except US)*: **ZAPPER TECHNOLOGIES INC.** [US/US]; Suite 501, 120 West 44 Street, New York, NY 10036 (US).

(72) Inventors; and

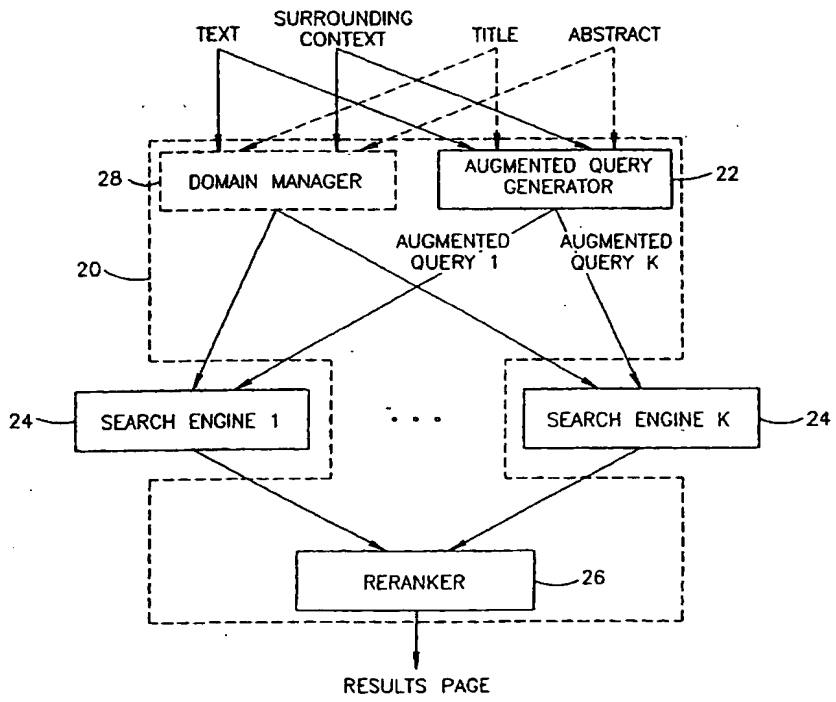
(75) Inventors/Applicants *(for US only)*: **RUPPIN, Eytan** [IL/IL]; 44 Lilach Street, 71908 Maccabim-Reut (IL). **FINKELSTEIN, Lev** [IL/IL]; 6/5 Zamenhof Street, 42309 Netanya (IL). **GABRILOVICH, Evgeniy** [IL/IL]; 46/2 Y.L. Baruch Street, 46323 Herzlia (IL). **SOLAN, Tsach** [IL/IL]; 10 Lea Street, 69412 Tel Aviv (IL). **RIVLIN, Ehud** [IL/IL]; 2 Lincoln Street, 34369 Haifa (IL). **MATIAS, Yosi** [IL/IL]; 12 Hamishmar Haezrachi, 69690 Tel Aviv (IL). **WOLFMAN, Gadi** [IL/IL]; 17 Balfur Street, 59311 Bat Yam (IL).

(74) Agent: **SINAI, Henry**; Eiran, Pearl, Latzer & Cohen-Zedek, 2 Gav Yam Center, 7 Shenkar Street, 46725 Herzlia (IL).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR,

[Continued on next page]

(54) Title: CONTEXT-DRIVEN INFORMATION RETRIEVAL



(57) Abstract: A system and method for retrieving and displaying search results includes an input unit, a generator, and a query manager. The input unit receives text for a query from a document and retrieves the context surrounding the text within the document. The generator (22) generates at least one augmented query to at least one search engine (24) using the text and the context. The query manager sends the augmented query to the search engine and retrieves the output of the search engine. The invention further includes a domain manager (28) connected to the input unit and to the query manager. The domain manager includes a domain selector for selecting a domain from a predetermined list of possible domains and a search engine selector for selecting the at least one search engine from a predetermined list of search engines associated with the selected domain. The invention also includes a reranker (26) connected to the query manager that receives at least one search result summary. The reranker includes at least one processor for generating at least one measure of similarity between the search results summaries and at least one of the text and the context and a sorter for ordering the search results summaries using the measures of similarity.

WO 01/31479 A1

LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) **Designated States** *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
-- *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

CONTEXT-DRIVEN INFORMATION RETRIEVAL

FIELD OF THE INVENTION

The present invention relates generally to computer-based information retrieval, and, in particular, to the utilization of context for the retrieval and display of textual material stored in files.

BACKGROUND OF THE INVENTION

The Internet is an unorganized mass of an overwhelming amount of information about a multitude of topics. A whole field has grown up around the topic of information retrieval (IR) in general and of search engines (SE) in particular, with the goal of making finding and retrieving information from the World Wide Web (Web) both faster and more accurate. First generation SEs are keyword based systems which use classical word vector methods to rank and retrieve documents, as a function of the frequency of appearance of queried words in the target documents. In other words, the frequency of query word appearance in a retrieved document is used to "rank" the document with a value from 0 - 100%. Documents are listed in the search results in rank order generally from highest to lowest. For an explanation of vector methods for search, see Kowalski, G., *Information Retrieval Systems: Theory and Implementation*, Ch. 5, Kluwer Academic Publishers, Boston, 1997. Unfortunately, such systems miss many sources of information that are relevant to the user's query and retrieve many unrelated and irrelevant documents.

Second generation SEs rely on link analysis and "popularity measurements", and essentially rank the retrieved documents by their "importance", in other words, by their "standing on the Web". Link analysis in search engines is described in J. Kleinberg, "Authoritative Sources in a Hyperlinked Environment", *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998 (extended version in the *Journal of the ACM* 46 1999). Popularity measurements are generated by the preferences of users, for example, by how often a site is visited. These SEs are limited since they do not find

"unimportant" documents, which may in fact be very relevant to the user's query even though their apparent importance may be low.

The current direction in IR research and development is a search technology that is capable of "understanding" the query and the target documents, and is able to retrieve the target documents in accordance to their semantic proximity to the query. Initial third generation SEs use the Latent Semantic Indexing algorithm, enabling the retrieval of matching documents even if they do not share any common words with the query. For example, an SE may try to find related keywords that were not in the original query. Latent Semantic Analysis (LSA) is described in S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Hirshman, "Indexing by Latent Semantic Analysis", *Journal of the American Society for Information Sciences*, vol. 41(6), pp. 391 - 407. More recently, third-generation SEs have attempted to utilize existing large semantic databases to distinguish among different senses of ambiguous concepts. A user is presented with a thesaurus-like list of synonyms that he must choose from to disambiguate words. However, such SEs depend on obtaining explicit feedback from the user by painstakingly querying him about the various possible meanings of the query.

SUMMARY OF THE INVENTION

An object of the present invention is to provide improved information retrieval from the Internet, by utilizing the context surrounding the query.

There is provided, in accordance with a preferred embodiment of the present invention, a system and method for the retrieval and display of search results. The system includes an input unit, a generator, and a query manager. The input unit receives text for a query from a document and retrieves the context surrounding the text within the document. The generator generates at least one augmented query to at least one search engine using the text and the context. The query manager sends the augmented query to the search engine and retrieves the output of the search engine.

Additionally, in accordance with a preferred embodiment of the present invention, the text includes at least one text word and the context includes at least one context word and the generator includes a manager and a query builder. The manager operates on the text and the context to select the generally more significant words therein and to provide importance indicators for each of the generally more significant words. The query builder creates the at least one query from the generally more significant words using the importance indicators.

Furthermore, in accordance with a preferred embodiment of the present invention, the manager includes any of a disambiguation estimator for checking the number of meanings of each of the words in the text and the context, a web frequency estimator, a proper name identifier for determining if the text word is a proper name, a synonymy gauge for counting the number of senses of the text word in at least one of a dictionary and a thesaurus, and a text gauge for counting the number of the words in the text. The web frequency estimator assigns a value corresponding to the frequency of the text word in a database of frequency of word usage and assigns a value to be used as input to the query builder.

Moreover, in accordance with a preferred embodiment of the present invention, the query builder includes a weight text operator, a decide include operator, a trim context operator, a weight context operator for determining repetitions of the context word, an augmented text builder, and an augmented context builder. The weight text operator uses the output of the web frequency

estimator and the disambiguation estimator for determining how many times to repeat the text word. The decide include operator uses the output of the proper name identifier and the synonymy gauge for determining if the word is prefixed with an include operator. The trim context operator uses the output of at least one of the text gauge, the synonymy gauge, the proper name identifier, and the disambiguation estimator all operative on the text for determining the amount of the context to use in the augmented query. The augmented text builder uses any of the output of the weight text operator and the decide include operator for constructing a text portion of the augmented query. The augmented context builder uses any of the output of the trim context operator and the weight context operator for constructing a context portion of the augmented query.

Moreover, in accordance with a preferred embodiment of the present invention, also including a manual control connected to the manager, the manual control is manipulatable by a user to increase or decrease the amount of the context retrieved by the input unit.

Further, in accordance with a preferred embodiment of the present invention, the query builder includes a weight text operator, a decide include operator, a weight context operator, an augmented text builder, and an augmented context builder. The weight text operator uses the output of the web frequency estimator and the disambiguation estimator for calculating how many times to repeat the word of text. The decide include operator uses the output of the proper name identifier and the synonymy gauge for determining if the word is prefixed with an include operator. The weight context operator is used for determining repetitions of the word of the context. The augmented text builder uses any results of the weight text operator and the decide include operator for constructing the text portion of the augmented query. The augmented context builder uses any results of the manual control and the weight context operator for constructing the context portion of the augmented query.

Additionally, in accordance with a preferred embodiment of the present invention, the system also includes a semantic network connected to the generator.

Furthermore, in accordance with a preferred embodiment of the present invention, the text includes at least one text word and the context includes at least one context word. The generator includes a cluster generator using the semantic network for generating a text cluster and at least one context cluster from the text and the context and a query builder for building at least one augmented query from the text cluster and the context cluster.

Moreover, in accordance with a preferred embodiment of the present invention, the cluster generator includes a text cluster generator and a context cluster generator.

Further, in accordance with a preferred embodiment of the present invention, the text cluster generator includes an apparatus for the comparing of each of the context words to generally every text word to find the average semantic distance of each of the context words to generally all of the text words and an apparatus for selecting for the text cluster at most X words whose average semantic distance is smaller than a predefined threshold.

Additionally, in accordance with a preferred embodiment of the present invention, the context cluster generator includes an apparatus for delineating between themes of the context words.

Furthermore, in accordance with a preferred embodiment of the present invention, the context includes the title of the document.

Moreover, in accordance with a preferred embodiment of the present invention, the context includes an abstract of the document.

Further, in accordance with a preferred embodiment of the present invention, also including a domain manager connected to the input unit and to the query manager, the domain manager includes a domain selector for selecting a domain from a predetermined list of possible domains and a search engine selector for selecting the at least one search engine from a predetermined list of search engines associated with the selected domain.

Additionally, in accordance with a preferred embodiment of the present invention, the domain selector includes a word relevancy estimator and a domain evaluator. The word relevancy estimator determines for each word in the text the probability of the word belonging to a domain. The domain evaluator determines

the normalized probability that the augmented query belongs to a domain using the word relevancy estimator results.

Furthermore, in accordance with a preferred embodiment of the present invention, the word relevancy estimator uses the following formula:

$$Score(Text \mid Domain_i) = \frac{\sum_{w_j \in Text} \frac{P(w_j \mid Domain_i)}{P'(w_j)}}{\#\ words\ in\ Text},$$

where $domain_i$ is an arbitrary domain i;

cohort refers to a general, non-domain specific information retrieval source;

w_j is an arbitrary word of said at least one text word j;

$$P'(w_i) = \begin{cases} P_{Cohort}(w_j), w_j \in Cohort \\ P(w_j), w_j \notin Cohort \end{cases};$$

$$P_{Cohort}(w_j) = P(w_j \mid Cohort);$$

$$P(w_j) = P(w_j \mid union\ of\ all\ domains);\ \text{and}$$

and where $P(w_j \mid Domain_i) = \frac{\#\ occurrences_{Domain_i}(w_j)}{\sum_{w_k \in Domain_i} \#\ occurrences_{Domain_i}(w_k)}$.

Moreover, in accordance with a preferred embodiment of the present invention, the domain evaluator uses the following formula:

$$F(x) = \frac{Y_{relevant}(x)}{Y_{relevant}(x) + prior * Y_{irrelevant}(x)}$$

where $Y_{relevant}(x)$ estimates the probability that the query belongs to the domain, and $Y_{irrelevant}(x)$ estimates the probability that the query does not belong to the domain, and prior is a constant relating to the probability that a random query belongs to a given domain.

Further in accordance with a preferred embodiment of the present invention, also including a reranker connected to the query manager and receiving at least one search result summary, the reranker includes at least one processor for generating at least one measure of similarity between the search results summaries and at least one of the text and the context and a sorter for ordering the search results summaries using the measures of similarity.

Additionally, in accordance with a preferred embodiment of the present invention, the measures of similarity are a boolean signifying whether or not generally all the text words are contained as a single phrase in the search result summary, the number of adjacent pairs of the text words appearing in the same form in the search result summary, and the number of adjacent pairs of the context words appearing in the same form in the search result summary.

Furthermore, in accordance with a preferred embodiment of the present invention, also including a semantic network connected to the reranker, the measures of similarity are the semantic distance from the text to the search result summary, the semantic distance from the context to the search result summary, the semantic distance from the search result summary to the text, and the semantic distance from the search result summary to the context.

Moreover, in accordance with a preferred embodiment of the present invention, the step of sending the augmented query to the search engine and retrieving the output of the search engine is sent over the Internet or an Intranet.

There is provided, in accordance with a preferred embodiment of the present invention, a system for the creation of a semantic network. The system includes a word estimator generator, a vector creator, a matrix creator, a distance determiner, and a matrix maker. The word estimator generator includes a word estimator and a set of K – 1 domain specific word estimators. The vector creator creates a K dimensional vector from generally each of the N words in the word estimator and the domain specific word estimators. The matrix creator forms an N x K matrix from the K dimensional vectors. The distance determiner determines the semantic proximity between generally each two words in the N x K matrix. The matrix maker creates an N x N matrix from the semantic proximities.

Additionally, in accordance with a preferred embodiment of the present invention, the word estimator generator includes an apparatus for selecting a set of random words, an apparatus for sending a query to a search engine including one of the random words, an apparatus for counting the number of occurrences of different words in each document returned, and an apparatus for creating a database including a list of the words and the number of occurrences of each of the words.

Moreover, in accordance with a preferred embodiment of the present invention, the apparatus for sending is done in a specific domain.

Finally, the present invention includes the methods performed by the system.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be understood and appreciated more fully from the following detailed description taken in conjunction with the appended drawings in which:

Figs. 1A and 1B are schematic illustrations of two documents with the same selected text but different surrounding contexts, useful in understanding the present invention;

Fig. 2 is a block diagram illustration of searching on the Internet in a context driven retrieval system, constructed and operative in accordance with a preferred embodiment of the present invention;

Figs. 3A and 3B are schematic illustrations of augmented queries constructed by the augmented query generator of Fig. 2;

Figs. 4A and 4B are schematic illustrations of the results of searches performed on the Internet using the augmented queries of Figs. 3A and 3B, which include rankings of the listed documents;

Fig. 5 is a block diagram illustration of the augmented query generator of Fig. 2, constructed and operative in accordance with an embodiment of the present invention;

Fig. 6 is a schematic illustration of the effects of step one of the filter of Fig. 5, constructed and operative in accordance with a preferred embodiment of the present invention;

Fig. 7 is a schematic illustration of the text and context manager of Fig. 5, constructed and operative in accordance with a preferred embodiment of the present invention;

Fig. 8 is a schematic illustration of the query builder of Fig. 5, constructed and operative in accordance with a preferred embodiment of the present invention;

Fig. 9 is a graphical illustration of a context-weighting curve;

Fig. 10 is a block diagram illustration of the detailed augmented query generator of Fig. 5 including a manual control for use by a user, constructed and operative in accordance with an alternative embodiment of the present invention;

Fig. 11 is a block diagram illustration the context driven retrieval system of Fig. 2 using a semantic network, constructed and operative in accordance with an alternative embodiment of the present invention;

Fig. 12 is a schematic illustration in two dimensions of a semantic network for the word "jaguar", useful in understanding the present invention;

Fig. 13 is a block diagram illustration of the augmented query generator of Fig. 11, constructed and operative in accordance with an alternative embodiment of the present invention;

Fig. 14 is a graphical illustration in two dimensions of the keywords clusters generated from the text and context of Fig.1A, by the context driven retrieval system of Fig. 11, constructed and operative in accordance with a preferred embodiment of the present invention;

Fig. 15 is a block diagram illustration of the domain manager of Fig. 2, constructed and operative in accordance with a preferred embodiment of the present invention; and

Fig. 16 is a block diagram illustration of the reranker of Fig. 2, constructed and operative in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PRESENT INVENTION

The present invention is a system and method for providing improved information retrieval from the Internet. Every search request has some context in which the query should be made. For example, words can have many different meanings, and the user will want a query made about a specific meaning. Applicants have realized that in searches based on a selection of text from within a document, the document itself contains context that can be used to disambiguate the search. The contents of the document, especially that part surrounding the selected portion, possibly along with the title and abstract (if one exists), provide context information. To obtain a more refined query, this context is used to limit the scope of the query. Applicants have further realized that use of such a query, augmented with context and limited in scope, results in a better search, that is, a search with fewer undesired results.

Reference is now made to Figs. 1A and 1B, schematic illustrations of two exemplary documents 10A and 10B with the same selected text 14, the word "jaguar", but otherwise generally different contents. It is clear to a reader of document 10A that it is about an animal, the jaguar. It is likewise clear, that document 10B is about a car, a Jaguar

Each document 10 has a title 12, selected text 14 that is underlined, and a surrounding context 16. Surrounding context 16 is a boxed-in section of document that surrounds text 14. Surrounding contexts 16A and 16B are generally completely different. Title 12A is "Jaguar is Often Confused With the Leopard". Title 12B is "The Consistently Expanding Market for Classic Jaguar".

Unfortunately, there is more than one meaning for the word "jaguar". Once the word is taken out of the context of the document, it is no longer clear which type of jaguar is being referred to and a search for word jaguar by itself would return many irrelevant references. In the case of a user reading document 10A, information about the animal jaguar is wanted. Whereas in the case of document 10B, what is wanted, is information about the car Jaguar. This is a common problem when formulating search requests.

Instead of the user having to add additional keywords to the search request, explicitly adding, for example, the words "car" or "animal", in a preferred

embodiment of the present invention, a predetermined number of the words surrounding text 14 are added automatically as surrounding context 16. These additional words are sent as an input to the search request generator in addition to text 14. This predetermined number of words to be extracted is the upper bound of the needed context and the search request generator uses the relevant parts of it. This enables a user simply to mark a section of document 10 as text 14, for example by highlighting, without the need for any further effort to clarify the search as required in the prior art.

It is also possible to add title 12 and an abstract if available, as further sources of context information. In a further embodiment of the present invention, they are also used.

US Patent Application 09/524,569, "Information Search and Retrieval System", filed on March 13, 2000 and assigned to the common assignee of the present invention, describes a system and method for selecting a word or words to use as the text input for a query request and for combining text and context for joint submission to a search engine, and is herein incorporated by reference.

US Provisional Patent Application 60/202,649 entitled "Text and Context Capture", filed on May 8, 2000 and assigned to the common assignee of the present invention, describes a system and method for capturing text and context words from text regions of applications for transfer and use in another application. These include applications running under any of the Microsoft Windows family of operating systems (available from Microsoft Corporation of Redmond, WA). US Provisional Patent Application 60/202,649 entitled "Text and Context Capture", filed on May 8, 2000 is herein incorporated by reference.

Fig. 2, to which reference is now made, is a block diagram illustration of searching on the Internet in a context driven retrieval system 20, constructed and operative in accordance with a preferred embodiment of the present invention. Context driven retrieval system 20 comprises an augmented query generator 22, a reranker 26, and an optional domain manager 28. At least one of any existing search engine 24 may be used with context driven retrieval system 20. Text 14, surrounding context 16, possibly title 12, and possibly an abstract are inputs to

augmented query generator 22 (herein referred to as "query generator") and to optional domain manager 28.

Query generator 22 takes its inputs and using the methods described hereinbelow in Figs. 5 - 9 and in Figs. 11 - 15 creates at least one "augmented query" (herein referred to as "augmented query" irrespective of the number of queries generated). Relevant words from the surrounding context 16, title 12 (if included), and the abstract (if included) are used in addition to the text in forming the query/queries. The resulting augmented query conforms to the syntax of search engine 24.

The output of query generator 22, an augmented query, is input to at least one search engine 24. Search engine 24 submits the augmented query to an information source and outputs the results. The results are then ordered by reranker 26 and output in a single ordered results page as described hereinbelow in Fig. 16.

In an alternative further embodiment of the present invention, text 14, surrounding context 16, possibly title 12, and possibly an abstract are input to domain manager 28. Domain manger 28 determines the topic(s) of its inputs and selects appropriate search engines 24 for the augmented query of query generator 22. At least one search engine 24 is selected per input topic and each augmented query generated in a topic is sent in parallel to all the search engines 24 selected for that topic, as described hereinbelow in Fig. 15.

Reference is now made to Figs. 3A and 3B, exemplary schematic illustrations of augmented queries constructed by augmented query generator 22. Augmented query 30 comprises an optional include operator 32, a weighted text 34, and a trimmed, weighted context 36. Fig. 3A shows an augmented query 30 for document 10A, whereas Fig. 3B shows an augmented query 30 for document 10B. Augmented query 30 of Figs. 3A and 3B is consistent with a typical Internet query.

When include operator 32 is used, the word following it must appear in the returned results. Weighted text 34 is the words of text 14 repeated as determined by query generator 22. Trimmed, weighted context contains selected words from surrounding context 16, possibly from title 12, and possibly from the

abstract. In Figs. 3A and 3B only surrounding context 16 is used. Only words considered significant are added to the query by query generator 22. Some words are added more than once, depending on their proximity to text 14.

Figs. 4A and 4B, to which reference is now made, are schematic illustrations of results pages of searches performed on the Internet using the augmented queries 30 of Figs. 3A and 3B, for documents 10A and 10B. A results page 37 consists of a plurality of references 38, wherein each reference 38 includes a ranking 40. Ranking 40 gives an indication of how well reference 38 matches the search criteria. It is clear when comparing Figs. 4A and 4B, that references 38 returned in Fig. 4A are completely different than those returned in Fig. 4B. References 38 of Fig. 4A are all about animals, whereas references 38 of Fig. 4B are all about cars.

Described hereinbelow are: an embodiment of augmented query generator 22 (Figs. 5 – 9); alternative embodiments of augmented query generator 22 of Fig. 5 (Fig. 10); an embodiment of augmented query generator 22 using a semantic network (Figs. 11 – 14); a preferred embodiment of domain manager 28 (Fig. 15); and a preferred embodiment of augmented reranker 26 (Fig. 16).

Augmented Query Generator 22 Using Heuristics

Reference is now made to Fig. 5, a block diagram illustration of augmented query generator 22, constructed and operative in accordance with one embodiment of the present invention, comprising a stop word and stemming filter 42 (herein referred to as filter), a text and context manager 44 (herein referred to as text manager), and a query builder 48. Text 14, surrounding context 16, optionally title 12, and optionally the abstract are each input to filter 42 for processing. Each grouping of words is processed separately and a filtered version is output. Filtered context herein refers to the combination of filtered surrounding context and, when available, filtered title and/or filtered abstract.

The filtered text and filtered context are input to text manger 44. As described hereinbelow in Fig. 7, text manager 44 processes the words, for example by duplicating some words. The resulting output of text manager 44 is

sent to query builder 48, which, as described hereinbelow in Fig. 8, combines the words into a single augmented query 30.

Fig. 6 is an exemplary schematic illustration of the effects on the surrounding context 16A (of Fig. 1A) of step one of filter 42, constructed and operative in accordance with a preferred embodiment of the present invention. In step one, filter 42 deletes certain words, so called "stop words", from text 14 and surrounding context 16A and produces text 14' and surrounding context 16A'. Filter 42 includes a predetermined list of words, which it removes from text 14 and surrounding context 16. See C. D. Manning, and H. Schutze, *Foundations of Statistical Natural Language Processing*, Ch. 15, MIT Press, Cambridge, MA, 1999 for an explanation of stop words and processing of natural languages. Comparing surrounding contexts 16A and 16A', it can be seen for example, that words "to", "a", "which", and "small" were deleted. Text 14 remains unchanged as there are no stop words in text 14. Filter 42, in a further step, takes the results of step one, for example surrounding context 16A', and removes a predetermined list of suffixes in a process called "stemming". Such a process is described in M. F. Porter, "Stemming - An Algorithm for Suffix Stripping", *Program* 14, pp. 130-137, 1980. Lastly, filter 42 removes all non-words. This filtering process is performed on all inputs to query generator 22.

Reference is now made to Fig. 7, a schematic illustration of text manager 44, constructed and operative in accordance with an embodiment of the present invention. Text manager 44 consists of a disambiguation estimator 52, a web frequency estimator 54, a proper name identifier 56, a synonymy gauge 58, and a text gauge 59. Text manager 44 receives the filtered text as input and works in conjunction with a thesaurus 53, a dictionary 55, and a web frequency database 57. Thesaurus 53 and dictionary 55 could be any commercially available products available for use on a computer. For example, Wordsmyth - The Educational Dictionary - Thesaurus available from Wordsmyth, Ithaca, N.Y. or http://www.wordsmyth.net/.

Disambiguation estimator 52 checks how many meanings or senses a single word i of filtered text can have. Each word is checked for the number of senses that appear for it in dictionary 55. This is done for every word i.

Disambiguation estimator 52 assigns these values to Di, giving the disambiguity of filtered text word i. In an alternative embodiment, thesaurus 53 is used instead of or in conjunction with dictionary 55.

Web frequency estimator 54 assigns a value Fi, which is the frequency with which filtered text word i appears on the World Wide Web ("Web"). This value is assigned from web frequency database 57, a static database of how frequently words appear on the Web. Applicants have developed a web frequency counter that samples the Web for the frequencies of each of a set of words. It submits a random set of queries to the Web, assembles the resulting documents, performs stop word and stemming filtering, counts each of the words appearing in the documents, and enters the word and its count into the database. It is further possible to have a domain specific set of queries, resulting in a domain specific word frequency database 57'. The Brown Corpus (maintained by the Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA) is an exemplary source of words for the web frequency counter.

Proper name identifier 56 performs several heuristics to determine if a filtered word i is a proper name and assigns the Boolean indicator Pi, one if yes zero if not. For example, it checks if word i is capitalized. It also checks both thesaurus 53 and dictionary 55 to determine that word i does not appear, as proper names are not listed in them.

Synonymy gauge 58 checks thesaurus 53 to count the number of senses that a word i of the filtered text has. It calculates the sum of all the synonyms over all the meanings of word i and assigns the value Si. For example, if a word has 2 meanings say, A and B, where A has two synonyms and B has one synonym, then Si = 3. In an alternative embodiment, dictionary 55 is used instead of or in conjunction with thesaurus 53.

Text gauge 59 counts the number of words in the filtered text and assigns the value to T. The greater the value of T, the more information the filtered text is assumed to contain, while a smaller T implies fewer words in the filtered text and hence less information.

Reference is now made to Figs. 8 and 9. Fig. 8 is a schematic illustration of query builder 48, constructed and operative in accordance with a preferred

embodiment of the present invention. Query builder 48 comprises a weight text operator 62, a decide include operator 64, a trim context operator 66, a weight context operator 68, an augmented text builder 70, an augmented context builder 72, and an augmented query builder 74.

The filtered text is used by both weight text operator 62 and decide include operator 64. The values Di and Fi of each word i are input to weight text operator 62. A calculation is made to determine how many times to repeat each word of the filtered text in the query, i.e., its relative weight. A smaller Fi means it appears with less frequency and, therefore, has more importance. Similarly the smaller Di is, the fewer meanings it has, and hence the word is given more weight. The occurencesTerm relates to Fi and the meaningsTerm relates to Di where:

$$occurencesTerm = \varepsilon * \left(1 - \frac{F_i}{thresholdF_i}\right) \text{ and}$$

$$meaningsTerm = \zeta * \left(1 - \frac{D_i}{thresholdD_i}\right)$$

Equation 1

where threshold Fi = max Fi value and threshold Di = max Di value.

The number of times the word is duplicated is the integer part (rounded value) of:

$$max\ Duplicates * (occurencesTerm + meaningsTerm)$$

where maxDuplicates is the predetermined value of the maximum allowed duplications of a single word and ε and ζ are predefined constants.

The values Pi and Si are input to decide include operator 64 and used to determine whether include operator 32 (Figs. 3A and 3B) should be added to the filtered text word. If a word is a proper name or its number of meanings Si is below a preset threshold, then include operator 32 is added before the word thus requiring its presence in any query result.

Augmented text builder 70 uses the outputs of weight text operator 62 and decide include operator 64 to construct the portion of the query resulting from the original text 14. If include operator 32 is used as indicated by decide include operator 64, then the word appears only once in the query. The word is not duplicated since the presence of the word in the result is guaranteed already,

implying maximum weighting of the word. If include operator 32 is not used, the word is duplicated according to weight text operator 62. All include operators 32 and word repetitions generated by all of the words i are concatenated to comprise the section of the query relating to text 14.

The filtered context is the input to both trim context operator 66 and weight context operator 68. Trim context operator 66 uses values T, S, P, and D generated over all the words i of filtered text to determine how much of the filtered context to actually use in the query. Note a set amount of surrounding context 16 is taken from document 10. However, if the meaning of the query is clear from the text itself then less filtered context is necessary. For example, if T is large, meaning that the filtered text length itself is long, then less filtered context is necessary since the meaning of the text selection is clearer. Alternatively, if D is small, less context is needed since there are few meanings to the text words. This "context factor" is given by:

$\alpha T + \beta S + \gamma P + \delta D$ where:

$$S = \frac{\sum Si}{T},$$

$$P = \frac{\sum Pi}{T},$$

$$D = \frac{\sum Di}{T}, \text{ and}$$

Equation 2

α, β, γ, and δ are positive constants whose values are predetermined by trial and error. Context factor is a fraction normalized to a value between 0 and 1. The normalized context factor is multiplied by the number of words in the trimmed context and rounded to an integer. This integer gives the number of words of context to use.

Weight context operator 68 uses a weighting curve to determine filtered context word repetitions. The closer a filtered context word is to text 14, the greater the weighting of the word. This weighting is described hereinbelow in Fig. 9 in greater detail.

Augmented context builder 72 uses the results of trim context operator 66 and weight context operator 68 to construct the portion of the query resulting from context 16. It takes the trimmed context and repeats word i according to its weighting.

The results of augmented text builder 70 and augmented context builder 72 are concatenated by augmented query builder 74 and the result is a single augmented query 30. Referring back to Fig. 3 for example, all the parts of an exemplary augmented query 30 can be seen.

Fig. 9 is a graphical illustration of a context weighting curve 76, used by weight context operator 68. The closer a word is to text 14 the greater its weight or duplication factor. Context weighting curve 76 always is greatest for a word next to a word of text 14. Context weighting curve 76 decreases slowly as the number of words from the marked text increases, so that words close to words of text 14 are still given a relatively high weight. However, it drops sharply and the weight factor becomes low above a distance of for example 10 - 15 words.

Augmented Query Generator 22 Including Manual Control

Reference is now made to Fig. 10, a block diagram illustration of the detailed augmented query generator 22 with the addition of a manual control for the user, constructed and operative in accordance with an alternative embodiment of the present invention. Augmented query generator 22 works in a similar manner to that described hereinabove in Fig. 5, similar elements are given similar reference numerals. However, a manual control 80 has been added, for example in a graphical user interface, so that the user can override the preset amount of surrounding context 16 automatically taken from document 10. This allows the user to try various refinements of the query using more or less surrounding context 16 in order to see how the query results are affected. Manual control 80 has levels which translate to a number between 0 and 1, valid context factor values. Using manual control 80, the user overrides the automatic calculation of the context factor. The manual control affects the number of times the text is duplicated as well as the length of the context. This data is used by query builder 48 instead of computing it as described hereinabove.

In another alternative embodiment of the present invention, a second disambiguation estimator 52 (not shown), to disambiguate context, is added to text and context manager 44. Receiving filtered context (optionally including context from the filtered title and filtered abstract) as input, the second disambiguation estimator 52 works as described hereinabove in Fig. 7, but on the words of the filtered context, and assigns each a value Dj. These values are used as further input to weight context operator 68.

Augmented Query Generator 22 Using A Semantic Network

Many public domain search engines (SE) divide information into different topics such as sports, news, health, and entertainment. These topic headings define the "domain" of the information. Furthermore, Applicants have realized that the development of semantic networks provides the basis for another framework for context use in context driven information retrieval. In another embodiment of the present invention, a semantic network is used in the generation of queries.

Fig. 11, to which reference is now made, is a block diagram illustration of searching on the Internet in a context driven retrieval system 20 similar to that of Fig. 2 but with the addition of a semantic network 110. Optional domain manager 28 is not shown. Similar objects are numbered similarly. Search engines 24 and reranker 26 operate as hereinabove and are not discussed further here.

Augmented queries generator 22 produces at least one augmented query 30 (Fig. 3). Generally, as described hereinbelow in Fig. 13, several queries are formed from each initial text 14, surrounding context 16, optional title 12, and optional abstract. Semantic network 110 is used to augment the queries with a more comprehensive list of semantically related keywords and hence to produce a more semantically focused search. It is further utilized to generate a series of augmented queries that are sent in parallel to a few target search engines.

In the human brain, words and concepts are stored in a multidimensional way with a feeling that certain words are "closer" to each other than others. For example, love and hate are closer than love and tomato. Close words are clustered together in multidimensional space. Semantic networks provide a computational way of generating semantic distances between words. Semantic

network 110 provides a knowledge base of values relating to the correlation between words and the word distribution within texts relating to the same topic. Semantic network 110 is created in a preprocessing stage and used to represent the multidimensional matrix space made up of the semantic distances between each two words. Each word in semantic network 110 is also represented by a K dimensional vector of its frequency of occurrence, in each of the K domains.

To create semantic network 110, a large directory on the World Wide Web that includes categorization is sampled to obtain a set of domain specific histograms, each describing the number of occurrences of all words appearing in the pertaining domain. These histograms are obtained by querying the search engine with a set of randomly generated single-word queries, collating the retrieved documents, and counting the number of occurrences of all words appearing in them. Thus, for each domain, a list of words and the frequency of the appearance of each word in the domain is created in a K dimensional vector. The total number of different words across all domains is N. If a word does not appear in a domain, it is given a frequency of 0 for that domain. After standard stop word filtering and stemming are performed, these N word vector representations are used to form an N x K occurrences matrix of words (rows) by domains (columns). There is always one domain of words in general, not related to a specific topic, and a predetermined number of content specific domains such as medicine or law.

Next, the semantic proximity between each two words in semantic network 110 is calculated to produce an N x N matrix of semantic distances. The semantic proximity between any two words X and Y is given by the correlation between their corresponding row vector representations,

$$P_{xy} = 1 - \frac{\text{Covariance}(\vec{x}, \vec{y})}{\sigma_x \sigma_y} = 1 - \frac{\sum_{d=1}^{k} (x_d - \bar{x})(y_d - \bar{y})}{\sqrt{\sum_{d=1}^{k} (x_d - \bar{x})^2 \sum_{d=1}^{k} (y_d - \bar{y})^2}}$$

Equation 3

where: $\vec{x}$ is the K dimensional vector of word x, $\bar{x}$ is the mean of word frequency values of word x, and σ is the standard deviation. See C. D. Manning, and H.

Schutze, *Foundations of Statistical Natural Language Processing*, Chapter 8, p. 300-301, MIT Press, Cambridge Massachusetts, 1999.

Fig. 12, to which reference is now made, is a two dimensional representation of an exemplary semantic network 110. There are many words connected to each other by lines, wherein such connections indicate that the connected words are related and the relative length of the lines give an indication of how strongly related the words are. The words car, jaguar, and animal are each surrounded by several words to which they are connected. The words in close proximity to these words form a cluster of related words. For example, bird, fur, body, tiger, jaguar, leopard, and elephant all form a cluster around animal. The word jaguar is related to the words leopard, animal, tiger, zoo, vehicle, car, garage, and plane. As can be seen by the length of the lines connecting the words in Fig. 12, jaguar has a much stronger relation to car and leopard than to garage or zoo.

Reference is now made to Fig. 13, a block diagram illustration of augmented query generator 22 comprising stop word and stemming filter 42, a text cluster generator 134, a context cluster generator 136, at least one context cluster 138, and a semantic augmented queries builder 140. Stop word and stemming filter 42 is the same as in Fig. 5 and is not described further here.

Text cluster generator 134 creates a cluster of a predetermined number of words, for example m, from the filtered context that are closest to the filtered text. A word in the filtered context is compared to every word in the text and the average value of the semantic distance is found. This is done for every word in the context. The m words with the closest average semantic distance are combined to create a text cluster, providing their distance is smaller than a predefined bound. The text cluster is appended to the filtered text by semantic augmented queries builder 140 to create a text oriented augmented query.

Context cluster generator 136 creates at least one context cluster 138 from the filtered context. The filtered context can contain several themes, which can lead to an unfocused query. Therefore, the invention delineates between themes to send a separate query dedicated to each theme separately. The K-means clustering algorithm is used to group the filtered context words into as

many significant context clusters 138 as it finds, up to a preset maximum, for example L. For an explanation of the K-means clustering algorithm, see C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK, 1995. Each context cluster is appended to the filtered text by semantic augmented queries builder 140, to create at least one and up to L, context oriented augmented queries.

Fig. 14, to which reference is now made, is a graphical illustration in two dimensions of the three exemplary keywords clusters formed from text 14 and surrounding context 16A of Fig. 1A. Text cluster includes the words: dots, body, tail, markings, shorter, small, yellow, and base. It is important to note that these words do not appear in the text but are now available to augmented queries generator 22 to help disambiguate the text meaning and form more focused queries. In this example, two context related clusters are created, herein labeled context cluster 1 and 2. Context cluster 1 includes the words: dots, color, body, small, and yellow. Context cluster 2 relates to a completely different topic than context cluster 1 and includes the words: irregular, larger, muscular, distinctively, presence, distinguished, fur, common, and shapes.

As can be seen in Fig. 14, text cluster and context cluster 1 overlap. It can also be seen that the words in a cluster are not always evenly distributed. It can further be seen that clusters need not overlap. Each context cluster relates to a different topic.

Domain Manager 28

Applicants have further realized that given the context of a query its domain can be determined. In a further preferred embodiment of the present invention, the domain of a query is established from among a preexisting list of domains. If an appropriate domain is found then searches are performed using a set of predefined domain specific search engines 24 (Fig. 2), each set containing at least one search engine 24. This results in a more focused query since search engines 24 specific to the domain are used. Unlike in the prior art, the selection is done by the system, without any need for user input. The query/queries are sent in parallel to all the search engines 24 contained in the set of search engines 24 for the domain.

Reference is now made to Fig. 15, a block diagram illustration of domain manager 28 (Fig. 2), comprising a domain selector 210, a domain search engine selector 220, a preexisting web frequency database 57, and a multiplicity of preexisting domain word frequency databases 57'. The keywords of text 14, surrounding context 16, optional title 12, and the optional abstract or augmented query 30 (Figs. 3A and 3B) are used as input. Augmented query 30 will be used hereinbelow in an exemplary fashion. The purpose of domain selector 210 is to choose which of the preexisting domains, if any, augmented query 30 belongs in. Augmented query 30 is constructed as described hereinabove.

Web frequency database 57 (Fig. 7) is as described hereinabove. Domain word frequency databases 57' are static databases constructed in a manner similar to web frequency database 57 but are domain specific. Thus, for each of the domains defined in domain selector 210 a static domain word frequency database 57' is created. In the creation of domain word frequency database 57', queries are sent using domain specific search engines 24, thus the words culled from the returned results contain words more generally found in the domain. Each domain word frequency database 57' is different, characterized by its own unique set of words and frequencies.

Domain selector 210 examines the word distribution in augmented query 30 and in its context and then by using domain word frequency databases 57' and web frequency database 57 determines which domain most closely reflects this word distribution. Once the search domain has been determined, domain search engine selector 220 submits augmented query 30 to each of the search engines 24 which were predefined for that domain. For example, if the domain is determined to be patents then two exemplary search engines 24 for the domain would be the Web sites of the U.S. Patents and Trademarks Office and the British Patent Office. These searches are done in parallel resulting in a series of results pages. These searches are performed in addition to the query sent to the general search engine 24.

Domain selector 210 consists of a word relevancy estimator 212 and a domain evaluator 214. Word relevancy estimator 212 uses web frequency database 57 and domain word frequency databases 57' and calculates a

probability score for each domain. Domain evaluator 214 uses the scores and performs calculations to select the search domain as described hereinbelow.

Word relevancy estimator 212 checks each word in the text and decides if the word is likely to be contained in a given domain, for example $domain_i$. This is done by comparing the frequency of the word in a domain, as given by domain word frequency database 57', to the frequency of the word in general, as given by web frequency database 57. For example, if a word has a low frequency in general but a high frequency in $domain_i$, then the presence of the word in the text indicates a likelihood that the domain required is $domain_i$. This comparison yields a probabilistic value for getting the correct domain.

Bayes' formula gives a way to combine aggregate probabilistic evidence correctly. The probability of a $domain_i$ being the correct domain given a particular text (for example augmented query 30), is $P(Domain_i|Text)$. This value according to Bayes' formula (refer to T. Mitchell, *Machine Learning*, Chapter 6, McGraw-Hill, New York, 1997) is:

$$P(Domain_i|Text) = \frac{P(Text|Domain_i) * P(Domain_i)}{P(Text)}.$$

Equation 4

The probability P(Text), the probability of the text in general, is the value given in web frequency database 57, a constant. In a preferred embodiment of the present invention, it is assumed that the probabilities of each of the domains being the subject of a given query is equal and hence $P(Domain_i)$ is a constant. Thus it remains to compute $P(Text|Domain_i)$, the probability of the text words appearing in $Domain_i$. In a preferred embodiment of the present invention, this probability is estimated as the sum of probabilities of all the text words w_j given $Domain_i$:

$$P(Text \mid Domain_i) = \sum_{w_j \in Text} P(w_j \mid Domain_i).$$

Equation 5

In the present invention this formula is extended. The extended formula produces a score for each domain and is referred to herein as the scoring function. Weighted word probabilities are used in order to increase the influence of important words. Additionally, the formula is normalized by the number of

words in the query text so that a longer text does not have a high score simply because many words were considered in the sum. The general frequency of a word $P'(w_j)$ is estimated using a cohort domain. In a preferred embodiment of the present invention, web frequency database 57 is used as the cohort domain. If the word is not included in web frequency database 57, then the cohort is generalized to be the union of all the supported domains and $P(w_j)$ is calculated instead. The probability of a word given a domain, $P(w_j|Domain_i)$, is the ratio of the number of occurrences of the word in the domain and the sum of the frequencies of all words in the domain. Thus:

$$Score(Text \mid Domain_i) = \frac{\sum_{w_j \in Text} \frac{P(w_j \mid Domain_i)}{P'(w_j)}}{\#\ words\ in\ Text}$$

where $$P'(w_j) = \begin{cases} P_{Cohort}(w_j), w_j \in Cohort \\ P(w_j), w_j \notin Cohort \end{cases}$$

and $$P_{Cohort}(w_j) = P(w_j \mid Cohort)$$

and $$P(w_j) = P(w_j \mid union\ of\ all\ domains)$$

and where $$P(w_j|Domain_i) = \frac{\#\ occurrences_{Domain_i}(w_j)}{\sum_{w_k \in Domain_i} \#\ occurrences_{Domain_i}(w_k)}.$$

Equation 6

Word relevancy estimator 212 gives each supported domain a score, for example x, where $x = P(Text|Domain_i)$. Since the scores in different domains have different ranges, it is necessary to normalize the score using a common scale to allow for comparison. Furthermore, it is necessary to have some confidence measure that the correct domain is chosen.

When choosing a domain there are four possible outcomes. The desired outcome is that the correct domain is chosen. However, the text may belong in an unsupported domain and hence any choice of domain would be wrong. The text may belong in a supported domain, but a different domain is chosen. Finally, the text may belong in a supported domain, but it is classified as not belonging to a supported domain. Simply picking the domain with the highest score would lead to increased misclassifications since a word can occur in $domain_i$ with less overall

frequency than in $domain_j$ and yet be more significant when it does appear in $domain_i$. Thus, a minimum frequency value is needed for every word in every domain.

This is done by domain evaluator 214 using an estimator function, F(x), that gives the normalized probability that augmented query 30 belongs to the domain. For every domain, a predefined constant threshold value T is calculated. For a classification of a query in a domain, for example $domain_j$, to be valid, in other words for the query to be relevant to $Domain_i$, its normalized probability must be greater than T_j. Thus, T_j is a confidence measure that the correct domain has been selected. The domain that is finally selected is the domain with the maximum normalized probability from among those domains whose threshold has been exceeded.

A query whose topic belongs to a given domain is "relevant" to the domain and all those not belonging are considered "irrelevant". For each domain, a model is created of the distribution of scores for relevant and irrelevant queries. This is done offline using a predefined set of queries which are sent to each domain, and for which sets of scores are calculated for each domain using the scoring function hereinabove. The distribution of scores is approximated using the Log-Normal distribution. Each distribution is fully defined by its mean (μ) and standard deviation (σ) and each domain model is represented by the combination of means and standard deviations of the relevant and irrelevant queries. Given x, the score value for which the distribution is calculated, the following pair of functions is used:

$$Y_{relevant}(x) = \text{Log-Normal}(x, \mu_{relevant}, \sigma_{relevant}) \text{ and}$$

$$Y_{irrelevant}(x) = \text{Log-Normal}(x, \mu_{irrelevant}, \sigma_{irrelevant}).$$

Equation 7

For each domain, a constant pairing of values $Y_{relevant}$ and $Y_{irrelevant}$ is created for all x.

For a query with score x as computed by word relevancy estimator 212, $Y_{relevant}(x)$ estimates the probability that the query belongs to the domain, whereas, $Y_{irrelevant}(x)$ estimates the probability that the query does not belong to the domain. The estimator function is defined as:

$$F(x) = \frac{Y_{relevant}(x)}{Y_{relevant}(x) + prior * Y_{irrelevant}(x)}$$

Equation 8

where prior is a constant relating to the probability that a random query belongs to a given domain.

Reranker 26

Fig. 16, to which reference is now made, is a block diagram illustration of reranker 26 (Fig. 2), constructed and operative in accordance with a preferred embodiment of the present invention. Reranker 26 comprises a C1 calculator 310, a C2 calculator 320, a C3 calculator 330, an optional C4-C7 calculator 340, a summary score calculator 350, and a summaries rescaler and reorderer 360.

The search performed by sending augmented query 30 to a target search engine 24 yields results page 37 (Figs. 4A and 4B) which includes a set of at least one summary. There may be many results pages, depending on how many queries were sent and to how many search engines 24 they were sent. Initially, each summary is filtered by filter 42 (Fig. 5) yielding filtered summaries. A matching algorithm measures the match between the retrieved summaries, and the original query and its pertaining context, to rerank the retrieved results and display the top ranked results to the user. The match between each result summary and the query is determined by measuring the mean pair wise semantic similarity between all the words in the summary and all the words in the text and context.

C1 calculator 310 assigns C1 to 1 if the full text string is contained in the summary and to 0 otherwise. For example, for the text "nice day", summary "really nice day" would have a C1 value of 1, whereas summary "day nice really" would have a C1 value of 0.

C2 calculator 320 checks the number of adjacent pairs of words from the text that appear in the same form in the summary, and assigns the value to C2. For example, for the text "really nice day", the summary "it is a really nice day" has C2 = 2, the summary "it is a nice day" has C2 = 1, and the summary "nice it is day really" has C2 = 0.

C3 calculator 330 counts the number of adjacent pairs of words from the context that appear in the same form in the summary and assigns the value to C3. This idea is similar to that of C2 using text, but will be given a lower overall weight than C2.

C4-C7 calculator 340 is optional and uses optional semantic network 110. If the embodiment contains semantic network 110, values C4, C5, C6, and C7 are calculated by C4-C7 calculator 340. These values are the semantic distances from the text to the summary (C4), from the context to the summary (C5), from the summary to the text (C6), and from the summary to the context (C7). Pairs C4 and C6, and C5 and C7 are necessary since semantic distances between words are not symmetrical.

Equation 3 describes the semantic proximity between two words. The semantic distance between two words is related to their semantic proximity. For clarity, the semantic distance calculation is described as well. The distance between two words w1 and w2 in a semantic network of K dimensions is defined as a correlation between their corresponding semantic vectors $w_1 = \langle w_{11} \cdots w_{1k} \rangle$ and $w_2 = \langle w_{21} \cdots w_{2k} \rangle$. Representing the mean of w_1 and w_2 as $\overline{w_1}$ and $\overline{w_2}$, and their standard deviations as σ_{w1} and σ_{w2}

$$dist(w_1, w_2) = \frac{\sum_{i=1}^{k} (w_{1i} - \overline{w_1})(w_{2i} - \overline{w_2})}{\sigma_{w1}\sigma_{w2}}.$$

Equation 9

The distance between a word w and a set of words S is defined as the minimum of the distances between w and the words of S. Thus if w is a text or context word then S is the set of summary words, and if w is a summary word then S is either the set of text or context words. Hence, given $w' \in S$

$$dist(w, S) = \min dist(w, w').$$

Equation 10

Finally, the asymmetric distance between two sets of words S_1 and S_2 is

$$dist(S_1, S_2) = \frac{1}{|S_1|} \sum_{w \in S_1} dist(w, S_2).$$

Equation 11

Hence, for C4 S_1 is the set of text words and S_2 is the set of summary words, for C5 S_1 is the set of context words and S_2 is the set of summary words, for C6 S_1 is the set of summary words and S_2 is the set of text words, and for C7 S_1 is the set of summary words and S_2 is the set of context words.

Summary score calculator 350 calculates a final score H for each of the summaries. Thus if there are Z summaries, Z scores H are calculated where the score of summary j is given by:

$$H_j = \sum_{i=1}^{7} \alpha_i C_i$$ if C4-C7 calculator 320 is used; and

$$H_j = \sum_{i=1}^{3} \alpha_i C_i$$ otherwise,

Equation 12

where α is a predefined constant. In a preferred embodiment of the present invention, $\alpha_1 = 10.0$, $\alpha_2 = 2.0$, $\alpha_3 = 1.0$, $\alpha_4 = 6.0$, $\alpha_5 = 4.0$, $\alpha_6 = 1.0$, and $\alpha_7 = 1.0$. A higher weight is thus given to text phrases found in the summaries (C1 and C2) and to text to summary (C4) and context to summary (C5) distances

Summaries rescaler and reorderer 360, takes the final scores H and normalizes them to a value between 0 and 100%. The summaries are sorted according to their final scores given by H_j and up to a predetermined number of the top ranking summaries, i.e. those with the highest relevance, are presented to the user on the results page.

It will be appreciated by persons skilled in the art that the present invention is not limited by what has been particularly shown and described hereinabove. Rather the scope of the invention is defined by the claims that follow:

CLAIMS

What is claimed is:

1. A system for the retrieval and display of search results, the system comprising:

an input unit for receiving text for a query from a document and for retrieving context surrounding said text within said document;

a generator for generating at least one augmented query to at least one search engine using said text and said context; and

a query manager for sending said at least one augmented query to said at least one search engine and retrieving the output of said at least one search engine.

2. A system according to claim 1 wherein said text includes at least one text word and said context includes at least one context word and wherein said generator comprises:

a manager operative on said text and said context for selecting the generally more significant text words and context words therein and for providing importance indicators for each of said generally more significant text words and context words; and

a query builder for creating said at least one query from said generally more significant text words and context words using said importance indicators.

3. A system according to claim 2 wherein said manager comprises any of:

a disambiguation estimator for checking the number of meanings of each of said at least one text word and context word;

a web frequency estimator for assigning a value corresponding to the frequency of said at least one text word in a database of frequency of word usage and assigning a value to be used as input to said query builder;

a proper name identifier for determining if said text word is a proper name;

a synonymy gauge for counting the number of senses of said text word in at least one of a dictionary and a thesaurus; and

a text gauge for counting the number of said at least one text word.

4. A system according to claim 3 wherein said query builder comprises:

a weight text operator using the output of said web frequency estimator and said disambiguation estimator for determining how many times to repeat said text word;

a decide include operator using the output of said proper name identifier and said synonymy gauge for determining if said text word is prefixed with an include operator;

a trim context operator using the output of at least one of said text gauge, said synonymy gauge, said proper name identifier, and said disambiguation estimator all operative on said text for determining the amount of said context to use in said augmented query;

a weight context operator for determining repetitions of said context word;

an augmented text builder using any of the output of said weight text operator and said decide include operator for constructing a text portion of said augmented query; and

an augmented context builder using any of the output of said trim context operator and said weight context operator for constructing a context portion of said augmented query.

5. A system according to claim 3 and also comprising a manual control connected to said manager, wherein said manual control is manipulatable by a user to increase or decrease the amount of said context retrieved by said input unit.

6. A system according to claim 5 and wherein said query builder comprises:

a weight text operator using the output of said web frequency estimator and said disambiguation estimator for calculating how many times to repeat said text word;

a decide include operator using the output of said proper name identifier and said synonymy gauge for determining if said text word is prefixed with an include operator;

a weight context operator for determining repetitions of said context word;

an augmented text builder using any results of said weight text operator and said decide include operator for constructing the text portion of said augmented query; and

an augmented context builder using any results of said manual control and said weight context operator for constructing the context portion of said augmented query.

7. A system according to claim 1 and wherein said system also comprises a semantic network connected to said generator.

8. A system according to claim 7 wherein said text includes at least one text word and said context includes at least one context word and wherein said generator comprises:

a cluster generator using said semantic network for generating a text cluster and at least one context cluster from said at least one text word and said at least one context word; and

a query builder for building at least one augmented query from said text cluster and said at least one context cluster.

9. A system according to claim 8 wherein said cluster generator comprises:

a text cluster generator for generating said text cluster from said at least one text word; and

a context cluster generator for generating said at least one context cluster from said at least one context word.

10. A system according to claim 9 wherein said text cluster generator comprises:

means for determining the average semantic distance of each of said at least one context words to generally all of said at least one text words; and

means for selecting for said text cluster at most X words whose average semantic distance is smaller than a predefined threshold.

11. A system according to claim 10 wherein said context cluster generator comprises means for delineating between themes of said context words.

12. A system according to claim 1 and wherein said context includes the title of said document.

13. A system according to claim 1 and wherein said context includes an abstract of said document.

14. A system according to claim 1 and also comprising a domain manager connected to said input unit and to said query manager, the domain manager comprising:

a domain selector for selecting a domain from a predetermined list of possible domains; and

a search engine selector for selecting said at least one search engine from a predetermined list of search engines associated with said selected domain.

15. A system according to claim 14 and wherein said domain selector comprises:

a word relevancy estimator which determines for each word in said text the probability of said word belonging to a domain; and

a domain evaluator which determines the normalized probability that said augmented query belongs to a domain using said word relevancy estimator results.

16. A system according to claim 15 and wherein said word relevancy estimator uses the following formula:

$$Score(Text \mid Domain_i) = \frac{\sum_{w_j \in Text} \frac{P(w_j \mid Domain_i)}{P'(w_j)}}{\#\ words\ in\ Text},$$

where $domain_i$ is an arbitrary domain i;

cohort refers to a general, non-domain specific information retrieval source;

w_j is an arbitrary word of said at least one text word j;

$$P'(w_j) = \begin{cases} P_{Cohort}(w_j), w_j \in Cohort \\ P(w_j), w_j \notin Cohort \end{cases};$$

$$P_{Cohort}(w_j) = P(w_j \mid Cohort);$$

$$P(w_j) = P(w_j \mid union\ of\ all\ domains);\ \text{and}$$

where $P(w_j|Domain_i) = \dfrac{\#\ occurrences_{Domain_i}(w_j)}{\sum_{w_k \in Domain_i} \#\ occurrences_{Domain_i}(w_k)}$.

17. A system according to claim 15 and wherein said domain evaluator uses the following formula:

$$F(x) = \frac{Y_{relevant}(x)}{Y_{relevant}(x) + prior * Y_{irrelevant}(x)}$$

where $Y_{relevant}(x)$ estimates the probability that the query belongs to the domain;

$Y_{irrelevant}(x)$ estimates the probability that the query does not belong to the domain; and

prior is a constant relating to the probability that a random query belongs to a given domain.

18. A system according to claim 1 and also comprising a reranker connected to said query manager and receiving at least one search result summary, the reranker comprising:

at least one processor for generating at least one measure of similarity between the at least one search results summary and at least one of said text and said context; and

a sorter for ordering said at least one search results summary using said at least one measure of similarity.

19. A system according to claim 18 and wherein said at least one measure of similarity is one of the following:

a boolean signifying whether or not generally all said text words are contained as a single phrase in the said search result summary;

the number of adjacent pairs of said text words appearing in the same form in said search result summary; and

the number of adjacent pairs of said context words appearing in the same form in said search result summary.

20. A system according to claim 18 and also comprising a semantic network connected to said reranker and wherein said at least one measure of similarity is one of the following:

a boolean signifying whether or not generally all said text words are contained as a single phrase in the said search result summary;

the number of adjacent pairs of said text words appearing in the same form in said search result summary;

the number of adjacent pairs of said context words appearing in the same form in said search result summary;

the semantic distance from said text to said search result summary;

the semantic distance from said context to said search result summary;

the semantic distance from said search result summary to said text; and

the semantic distance from said search result summary to said context.

21. A system for the creation of a semantic network the system comprising:

a word estimator generator comprising:

a general word estimator for creating a general database of words and their frequencies; and

a set of K – 1 domain specific word estimators for creating a domain specific databases of words and their frequencies;

a matrix vector creator for creating an N x K matrix from N K dimensional vectors wherein said K dimensional vectors are created from generally each of the N words in said word estimator and said domain specific word estimators;

a distance determiner for determining the semantic proximity between at least one pair of words in said N x K matrix; and

a matrix maker for creating an N x N matrix from said semantic proximities.

22. A system according to claim 21 and wherein said word estimator generator comprises:

means for selecting a set of random words;

means for sending a query to a search engine, the query comprising one of said random words;

means for counting the number of occurrences of different words in each document returned; and

means for creating a database comprising a list of said words and said number of occurrences of each of said words.

23. A system according to claim 22 and wherein said means for sending operates in a specific domain.

24. A method for the retrieval and display of search results, the method comprising the steps of:

receiving text for a query from a document;

retrieving context surrounding said text within said document;

generating at least one augmented query to at least one search engine using said text and said context; and

sending said at least one augmented query to said at least one search engine and retrieving the output of said at least one search engine.

25. A method according to claim 24 wherein said text includes at least one text word and said context includes at least one context word and wherein said generating step includes the steps of:

selecting from said text and said context the generally more significant words therein;

providing importance indicators for each of said generally more significant words; and

creating said at least one query from said generally more significant words using said importance indicators.

26. A method according to claim 25 wherein said selecting step and said providing step include the steps of:

checking the number of meanings of each of the words in said text and said context;

assigning a value corresponding to the frequency of said text word in a database of frequency of word usage;

concluding if said text word is a proper name;

measuring the number of senses of said text word in at least one of a dictionary and a thesaurus; and

counting the number of said text words.

27. A method according to claim 26 wherein said creating step includes the steps of:

ascertaining how many times to repeat said text word using the output of said steps of assigning and checking;

deciding if said word is prefixed with an include operator using the output of said steps of concluding and measuring;

determining the amount of said context to use in said augmented query using the output of at least one of said steps of counting, measuring, concluding, and checking all operative on said text;

setting repetitions of said context word;

constructing a text portion of said augmented query using any of the output of said steps of ascertaining and deciding; and

constructing a context portion of said augmented query using any of the output of said steps of determining and setting.

28. A method according to claim 25 and also including the step of reading the settings on a manual control, wherein said manual control is manipulatable by a user to increase or decrease the amount of said context retrieved by said retrieving step.

29. A method according to claim 28 and wherein said creating step includes the steps of:

ascertaining how many times to repeat said text word using the output of said steps of assigning and checking;

deciding if said word is prefixed with an include operator using the output of said steps of concluding and measuring;

determining the amount of said context to use in said augmented query using the output of at least one of said steps of counting, measuring, concluding, and checking all operative on said text;

setting repetitions of said context word;

constructing a text portion of said augmented query using any of the output of said steps of ascertaining and deciding; and

constructing a context portion of said augmented query using any of the output of said steps of reading the settings on a manual control and setting.

30. A method according to claim 24 wherein said text includes at least one text word and said context includes at least one context word and wherein said step of generating includes the steps of:

creating a text cluster from said text and said context using a semantic network;

producing at least one context cluster from said context using said semantic network; and

building at least one augmented query from said text cluster and said at least one context cluster in a query builder.

31. A method according to claim 30 wherein said step of creating includes the steps of:

determining the average semantic distance of each of said context words to generally all said text words; and

selecting for said text cluster at most X words whose average semantic distance is smaller than a predefined threshold.

32. A method according to claim 30 wherein said step of producing includes the step of delineating between themes of said context words.

33. A method according to claim 24 and wherein said step of retrieving includes the step of retrieving the title of said document.

34. A method according to claim 24 and wherein said step of retrieving includes the step of retrieving an abstract of said document.

35. A method according to claim 24 and wherein said method includes the steps of:

selecting a domain from a predetermined list of possible domains; and

choosing said at least one search engine from a predetermined list of search engines associated with said selected domain.

36. A method according to claim 35 and wherein said step of selecting includes the steps of:

determining for each word in said text the probability of said word belonging to a domain; and

ascertaining the normalized probability that said augmented query belongs to a domain using results of said step of determining.

37. A method according to claim 36 and wherein said step of determining includes the step of computing the following formula:

$$Score(Text \mid Domain_i) = \frac{\sum_{w_j \in Text} \frac{P(w_j \mid Domain_i)}{P'(w_j)}}{\#\ words\ in\ Text},$$

where $domain_i$ is an arbitrary domain i;

cohort refers to a general, non-domain specific information retrieval source;

w_j is an arbitrary word of said at least one text word j;

$$P'(w_i) = \begin{cases} P_{Cohort}(w_j), w_j \in Cohort \\ P(w_j), w_j \notin Cohort \end{cases};$$

$$P_{Cohort}(w_j) = P(w_j \mid Cohort);$$

$$P(w_j) = P(w_j \mid union\ of\ all\ domains);\ \text{and}$$

$$\text{where } P(w_j \mid Domain_i) = \frac{\#\ occurrences_{Domain_i}(w_j)}{\sum_{w_k \in Domain_i} \#\ occurrences_{Domain_i}(w_k)}.$$

38. A method according to claim 36 and wherein said a step of ascertaining includes the step of computing the following formula:

$$F(x) = \frac{Y_{relevant}(x)}{Y_{relevant}(x) + prior * Y_{irrelevant}(x)}$$

where $Y_{relevant}(x)$ estimates the probability that the query belongs to the domain;

$Y_{irrelevant}(x)$ estimates the probability that the query does not belong to the domain; and

prior is a constant relating to the probability that a random query belongs to a given domain.

39. A method according to claim 24 and also including the steps of:

computing at least one measure of similarity between the at least one search results summary and at least one of said text and said context; and

ordering said at least one search results summary using said at least one measure of similarity.

40. A method according to claim 39 and wherein said at least one measure of similarity is one of the following:

a boolean signifying whether or not generally all said text words are contained as a single phrase in the said search result summary;

the number of adjacent pairs of said text words appearing in the same form in said search result summary; and

the number of adjacent pairs of said context words appearing in the same form in said search result summary.

41. A method according to claim 39 and wherein said wherein said at least one measure of similarity is one of the following:

a boolean signifying whether or not generally all said text words are contained as a single phrase in the said search result summary;

the number of adjacent pairs of said text words appearing in the same form in said search result summary;

the number of adjacent pairs of said context words appearing in the same form in said search result summary;

the semantic distance from said text to said search result summary;

the semantic distance from said context to said search result summary;

the semantic distance from said search result summary to said text; and

the semantic distance from said search result summary to said context.

42. A method according to claim 24 wherein said step of sending and said step of retrieving are performed over the Internet.

43. A method according to claim 24 wherein said step of sending and said step of retrieving are performed over the Intranet.

44. A method for the creation of a semantic network the method including the steps of:

creating a general word estimator and a set of K – 1 domain specific word estimators;

representing as a K dimensional vector each of N words in said word estimator and said domain specific word estimators;

forming an N x K matrix from said K dimensional vectors ;

determining the semantic proximity between each two words in said N x K matrix; and

making an N x N matrix from said semantic proximities.

45. A method according to claim 44 and wherein said step of creating includes the steps of:

selecting a set of random words;

sending a query to a search engine, the query comprising one of said random words;

counting the number of occurrences of different words in each document returned; and

creating a database comprising a list of said words and said number of occurrences of each of said words.

46. A method according to claim 45 and wherein said step of sending is performed for a specific domain.

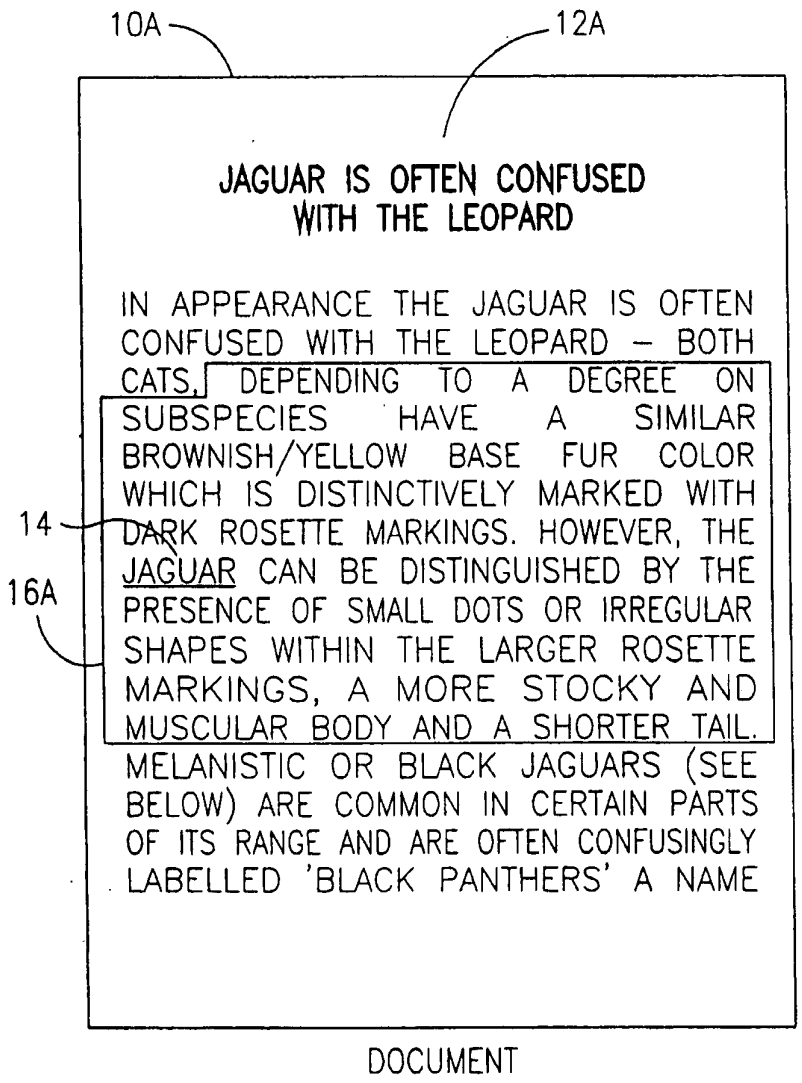

10A
12A
JAGUAR IS OFTEN CONFUSED WITH THE LEOPARD
IN APPEARANCE THE JAGUAR IS OFTEN CONFUSED WITH THE LEOPARD – BOTH CATS, DEPENDING TO A DEGREE ON SUBSPECIES HAVE A SIMILAR BROWNISH/YELLOW BASE FUR COLOR WHICH IS DISTINCTIVELY MARKED WITH DARK ROSETTE MARKINGS. HOWEVER, THE JAGUAR CAN BE DISTINGUISHED BY THE PRESENCE OF SMALL DOTS OR IRREGULAR SHAPES WITHIN THE LARGER ROSETTE MARKINGS, A MORE STOCKY AND MUSCULAR BODY AND A SHORTER TAIL. MELANISTIC OR BLACK JAGUARS (SEE BELOW) ARE COMMON IN CERTAIN PARTS OF ITS RANGE AND ARE OFTEN CONFUSINGLY LABELLED 'BLACK PANTHERS' A NAME
14
16A
DOCUMENT

FIG.1A

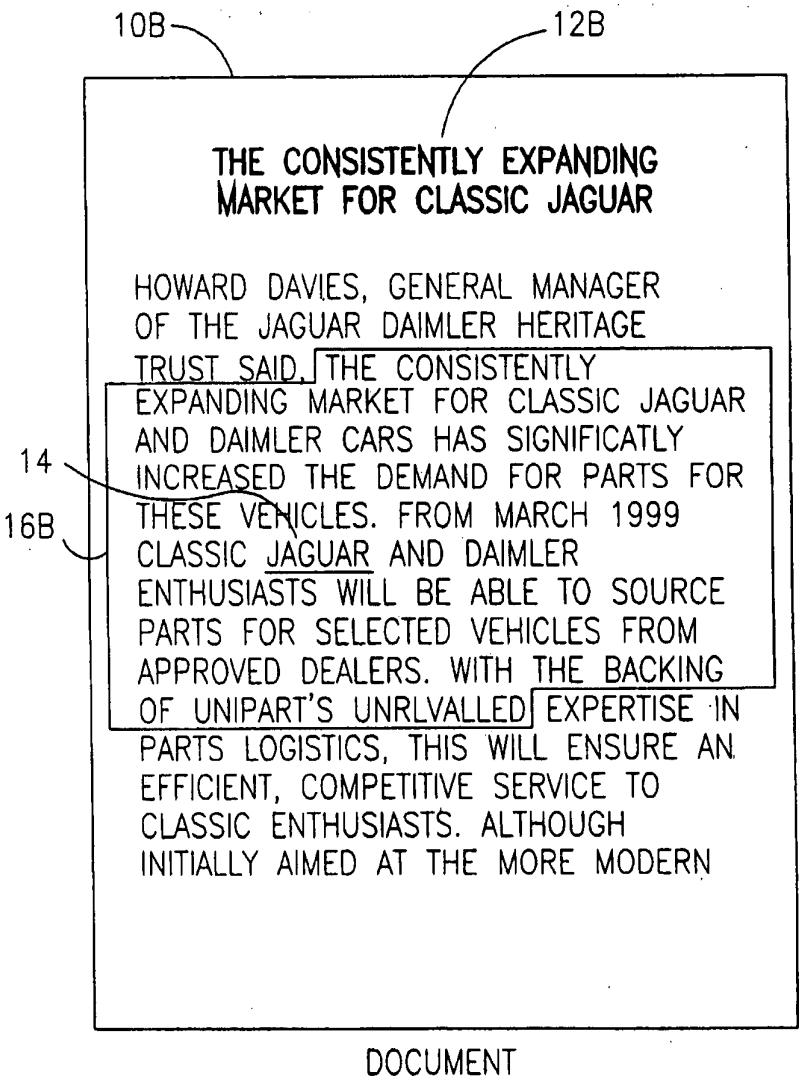

10B
12B
THE CONSISTENTLY EXPANDING MARKET FOR CLASSIC JAGUAR
HOWARD DAVIES, GENERAL MANAGER OF THE JAGUAR DAIMLER HERITAGE TRUST SAID, THE CONSISTENTLY EXPANDING MARKET FOR CLASSIC JAGUAR AND DAIMLER CARS HAS SIGNIFICATLY INCREASED THE DEMAND FOR PARTS FOR THESE VEHICLES. FROM MARCH 1999 CLASSIC JAGUAR AND DAIMLER ENTHUSIASTS WILL BE ABLE TO SOURCE PARTS FOR SELECTED VEHICLES FROM APPROVED DEALERS. WITH THE BACKING OF UNIPART'S UNRLVALLED EXPERTISE IN PARTS LOGISTICS, THIS WILL ENSURE AN EFFICIENT, COMPETITIVE SERVICE TO CLASSIC ENTHUSIASTS. ALTHOUGH INITIALLY AIMED AT THE MORE MODERN
14
16B
DOCUMENT

FIG.1B

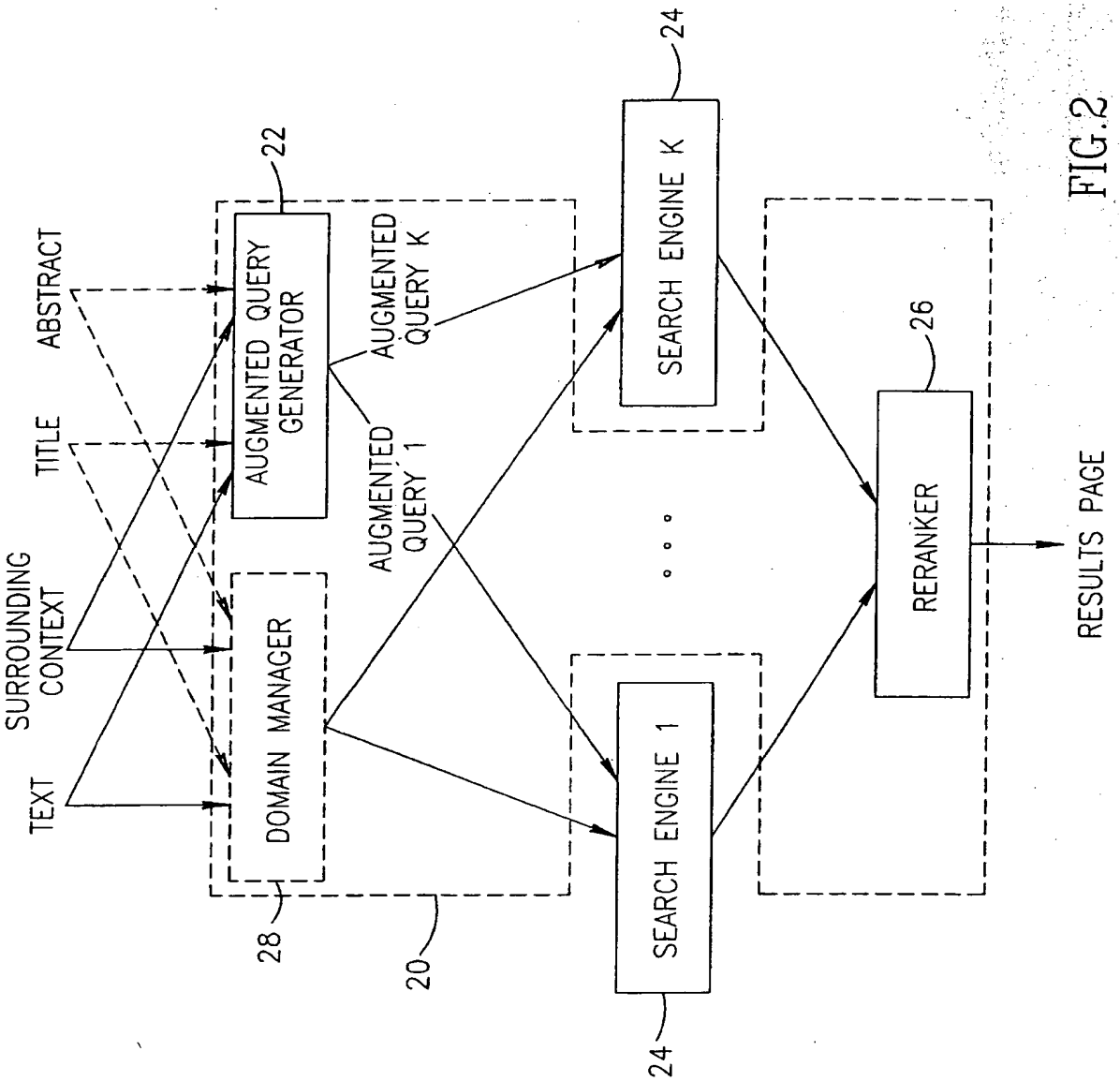

ABSTRACT
TITLE
SURROUNDING CONTEXT
TEXT
AUGMENTED QUERY GENERATOR
22
DOMAIN MANAGER
28
AUGMENTED QUERY K
AUGMENTED QUERY 1
20
SEARCH ENGINE K
24
SEARCH ENGINE 1
24
RERANKER
26
RESULTS PAGE

FIG.2

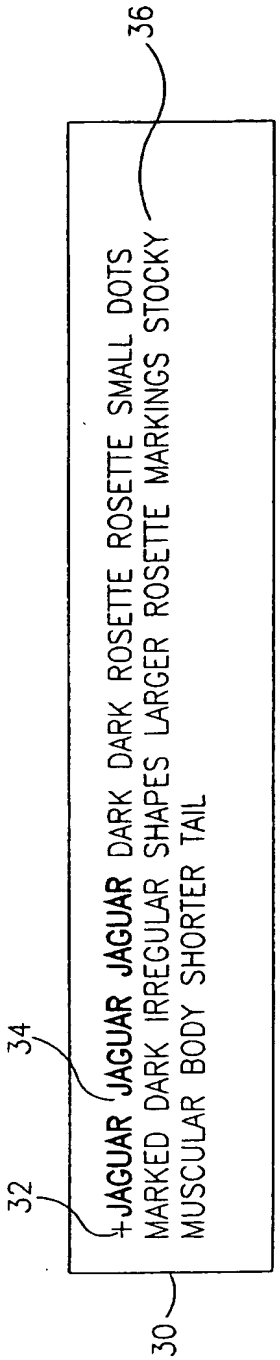

30
32
34
36
+JAGUAR JAGUAR DARK DARK ROSETTE ROSETTE SMALL DOTS
MARKED DARK IRREGULAR SHAPES LARGER ROSETTE MARKINGS STOCKY
MUSCULAR BODY SHORTER TAIL

FIG.3A

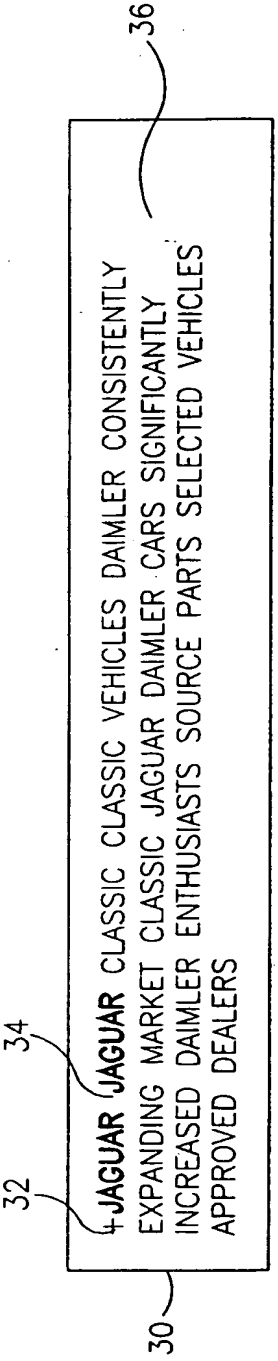

30
32
34
36
+JAGUAR JAGUAR CLASSIC CLASSIC VEHICLES DAIMLER CONSISTENTLY
EXPANDING MARKET CLASSIC JAGUAR DAIMLER CARS SIGNIFICANTLY
INCREASED DAIMLER ENTHUSIASTS SOURCE PARTS SELECTED VEHICLES
APPROVED DEALERS

FIG.3B

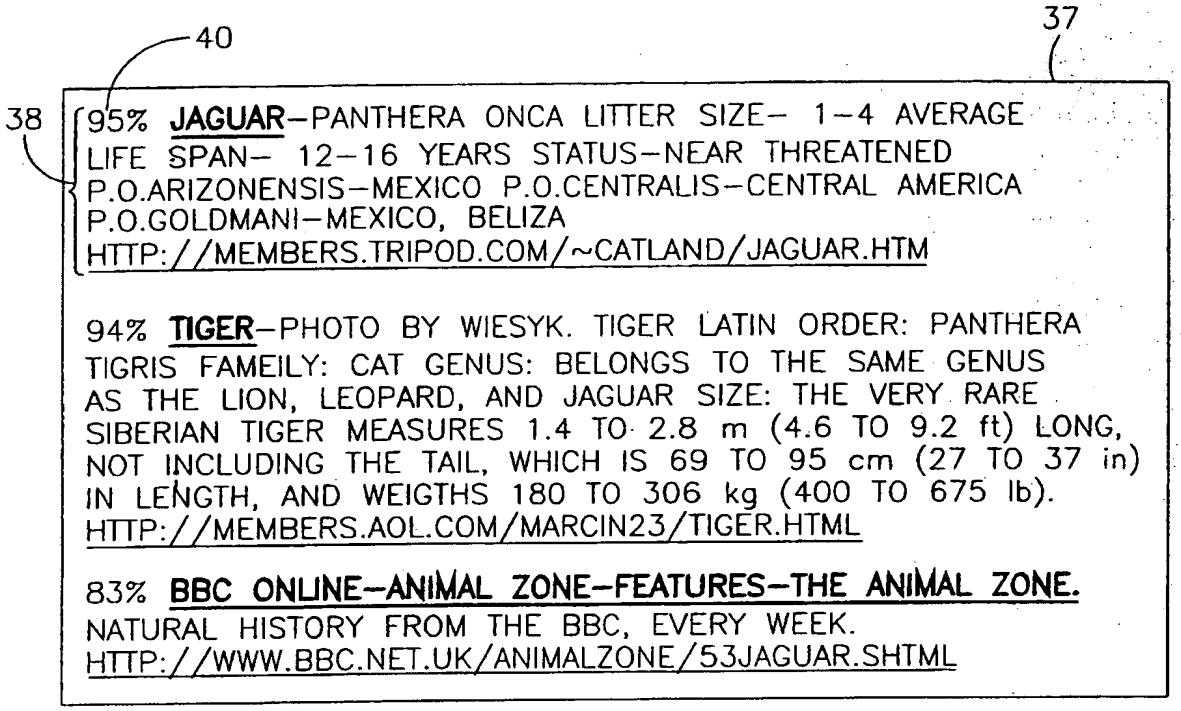

37
40
38
95% JAGUAR–PANTHERA ONCA LITTER SIZE– 1–4 AVERAGE LIFE SPAN– 12–16 YEARS STATUS–NEAR THREATENED P.O.ARIZONENSIS–MEXICO P.O.CENTRALIS–CENTRAL AMERICA P.O.GOLDMANI–MEXICO, BELIZA
HTTP://MEMBERS.TRIPOD.COM/~CATLAND/JAGUAR.HTM
94% TIGER–PHOTO BY WIESYK. TIGER LATIN ORDER: PANTHERA TIGRIS FAMEILY: CAT GENUS: BELONGS TO THE SAME GENUS AS THE LION, LEOPARD, AND JAGUAR SIZE: THE VERY RARE SIBERIAN TIGER MEASURES 1.4 TO 2.8 m (4.6 TO 9.2 ft) LONG, NOT INCLUDING THE TAIL, WHICH IS 69 TO 95 cm (27 TO 37 in) IN LENGTH, AND WEIGTHS 180 TO 306 kg (400 TO 675 lb).
HTTP://MEMBERS.AOL.COM/MARCIN23/TIGER.HTML
83% BBC ONLINE–ANIMAL ZONE–FEATURES–THE ANIMAL ZONE.
NATURAL HISTORY FROM THE BBC, EVERY WEEK.
HTTP://WWW.BBC.NET.UK/ANIMALZONE/53JAGUAR.SHTML

FIG.4A

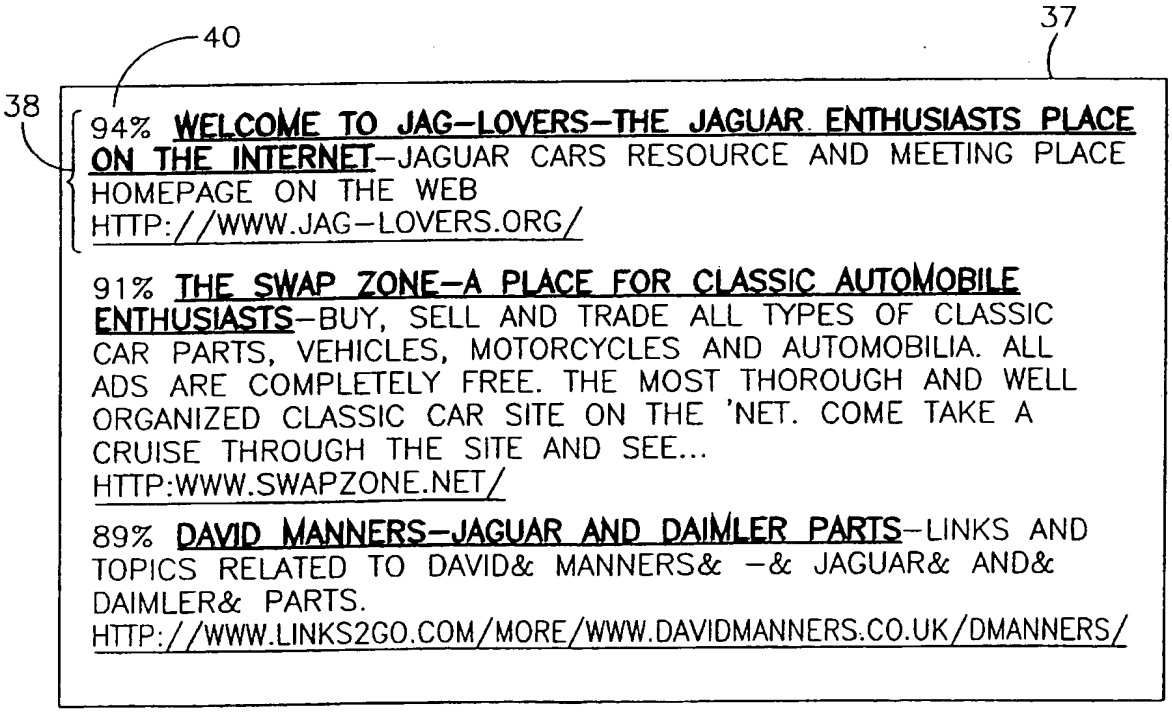

37
40
38
94% WELCOME TO JAG–LOVERS–THE JAGUAR ENTHUSIASTS PLACE ON THE INTERNET–JAGUAR CARS RESOURCE AND MEETING PLACE HOMEPAGE ON THE WEB
HTTP://WWW.JAG–LOVERS.ORG/
91% THE SWAP ZONE–A PLACE FOR CLASSIC AUTOMOBILE ENTHUSIASTS–BUY, SELL AND TRADE ALL TYPES OF CLASSIC CAR PARTS, VEHICLES, MOTORCYCLES AND AUTOMOBILIA. ALL ADS ARE COMPLETELY FREE. THE MOST THOROUGH AND WELL ORGANIZED CLASSIC CAR SITE ON THE 'NET. COME TAKE A CRUISE THROUGH THE SITE AND SEE...
HTTP:WWW.SWAPZONE.NET/
89% DAVID MANNERS–JAGUAR AND DAIMLER PARTS–LINKS AND TOPICS RELATED TO DAVID& MANNERS& –& JAGUAR& AND& DAIMLER& PARTS.
HTTP://WWW.LINKS2GO.COM/MORE/WWW.DAVIDMANNERS.CO.UK/DMANNERS/

FIG.4B

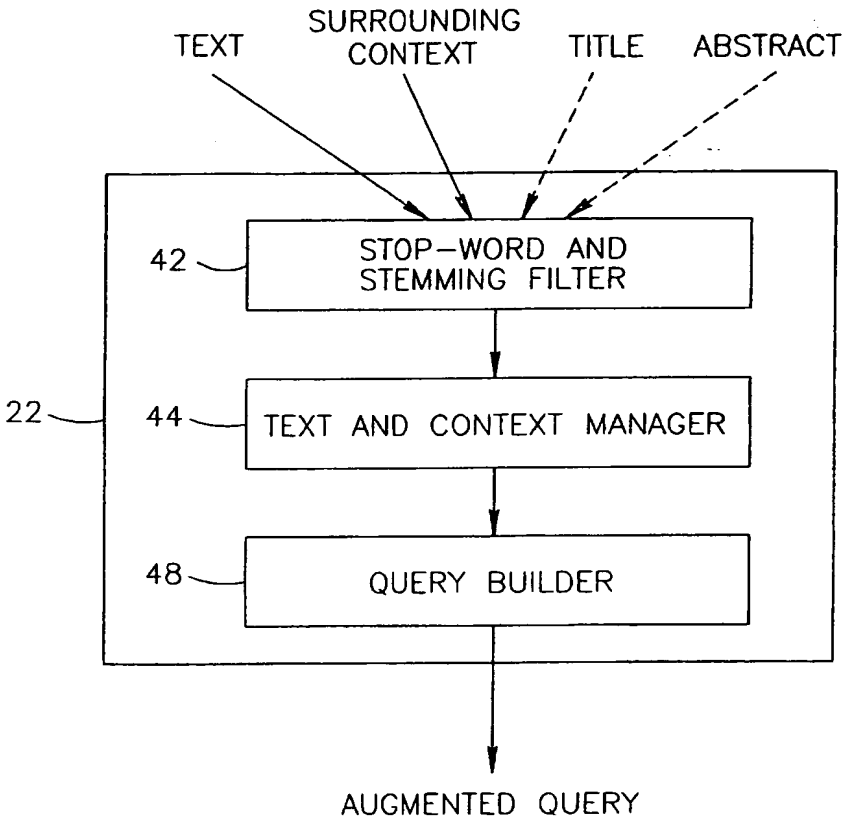

TEXT
SURROUNDING CONTEXT
TITLE
ABSTRACT
42
STOP–WORD AND STEMMING FILTER
22
44
TEXT AND CONTEXT MANAGER
48
QUERY BUILDER
AUGMENTED QUERY

FIG.5

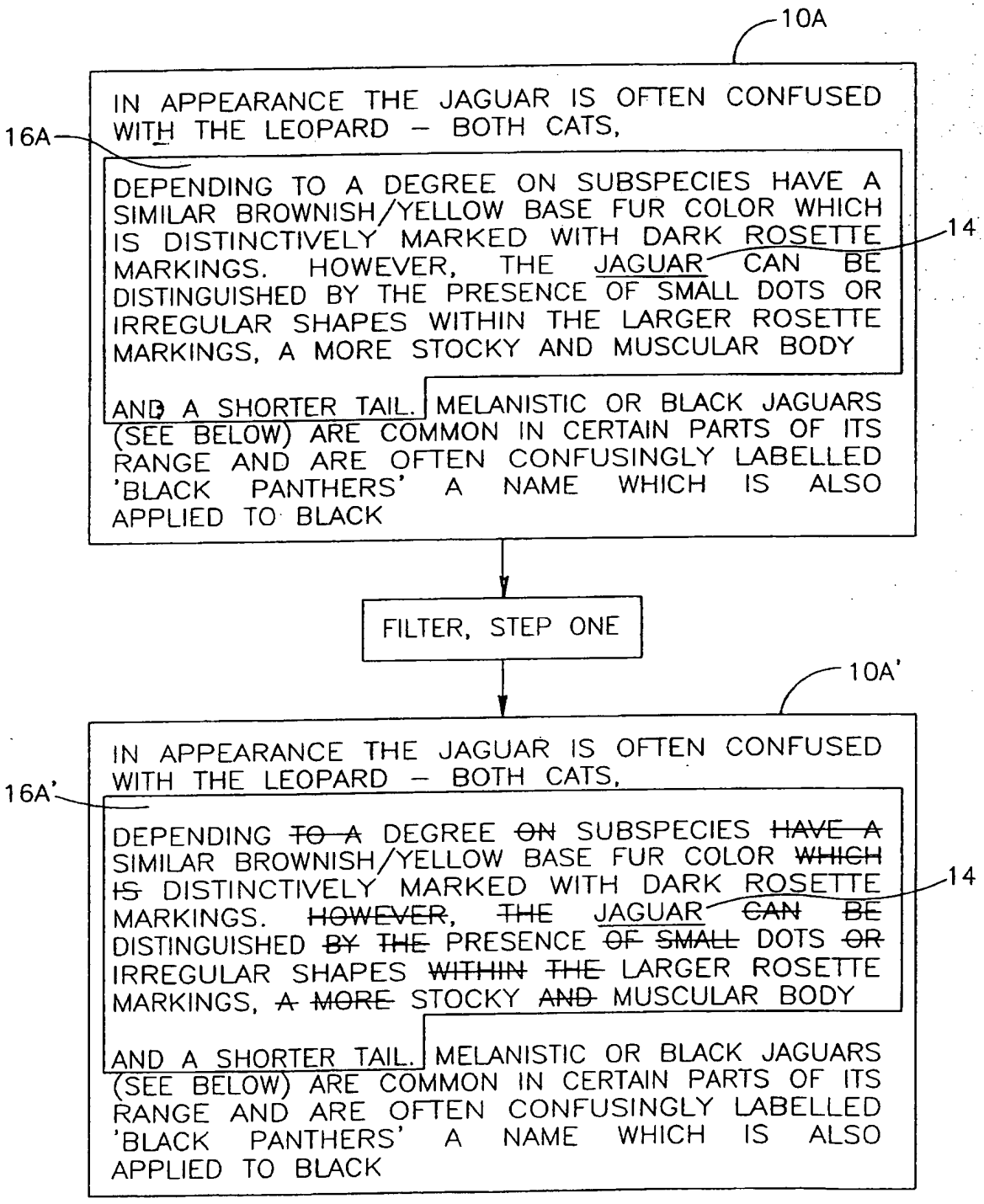

10A
IN APPEARANCE THE JAGUAR IS OFTEN CONFUSED WITH THE LEOPARD – BOTH CATS,
16A
DEPENDING TO A DEGREE ON SUBSPECIES HAVE A SIMILAR BROWNISH/YELLOW BASE FUR COLOR WHICH IS DISTINCTIVELY MARKED WITH DARK ROSETTE MARKINGS. HOWEVER, THE JAGUAR CAN BE DISTINGUISHED BY THE PRESENCE OF SMALL DOTS OR IRREGULAR SHAPES WITHIN THE LARGER ROSETTE MARKINGS, A MORE STOCKY AND MUSCULAR BODY
14
AND A SHORTER TAIL. MELANISTIC OR BLACK JAGUARS (SEE BELOW) ARE COMMON IN CERTAIN PARTS OF ITS RANGE AND ARE OFTEN CONFUSINGLY LABELLED 'BLACK PANTHERS' A NAME WHICH IS ALSO APPLIED TO BLACK
FILTER, STEP ONE
10A'
IN APPEARANCE THE JAGUAR IS OFTEN CONFUSED WITH THE LEOPARD – BOTH CATS,
16A'
DEPENDING TO A DEGREE ON SUBSPECIES HAVE A SIMILAR BROWNISH/YELLOW BASE FUR COLOR WHICH IS DISTINCTIVELY MARKED WITH DARK ROSETTE MARKINGS. HOWEVER, THE JAGUAR CAN BE DISTINGUISHED BY THE PRESENCE OF SMALL DOTS OR IRREGULAR SHAPES WITHIN THE LARGER ROSETTE MARKINGS, A MORE STOCKY AND MUSCULAR BODY
14
AND A SHORTER TAIL. MELANISTIC OR BLACK JAGUARS (SEE BELOW) ARE COMMON IN CERTAIN PARTS OF ITS RANGE AND ARE OFTEN CONFUSINGLY LABELLED 'BLACK PANTHERS' A NAME WHICH IS ALSO APPLIED TO BLACK

FIG.6

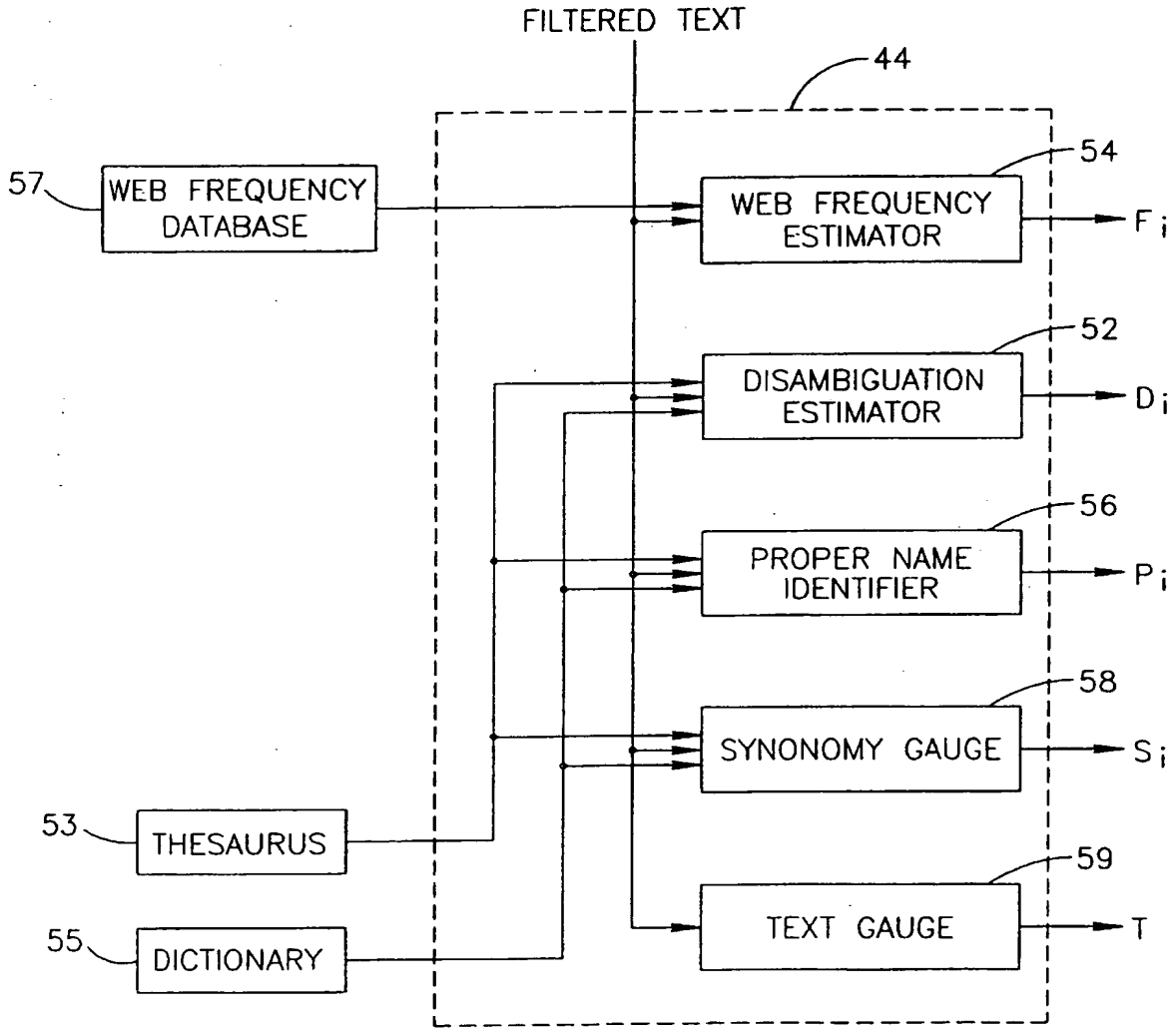

FILTERED TEXT
44
57
WEB FREQUENCY DATABASE
54
WEB FREQUENCY ESTIMATOR
Fi
52
DISAMBIGUATION ESTIMATOR
Di
56
PROPER NAME IDENTIFIER
Pi
58
SYNONOMY GAUGE
Si
59
TEXT GAUGE
T
53
THESAURUS
55
DICTIONARY

FIG.7

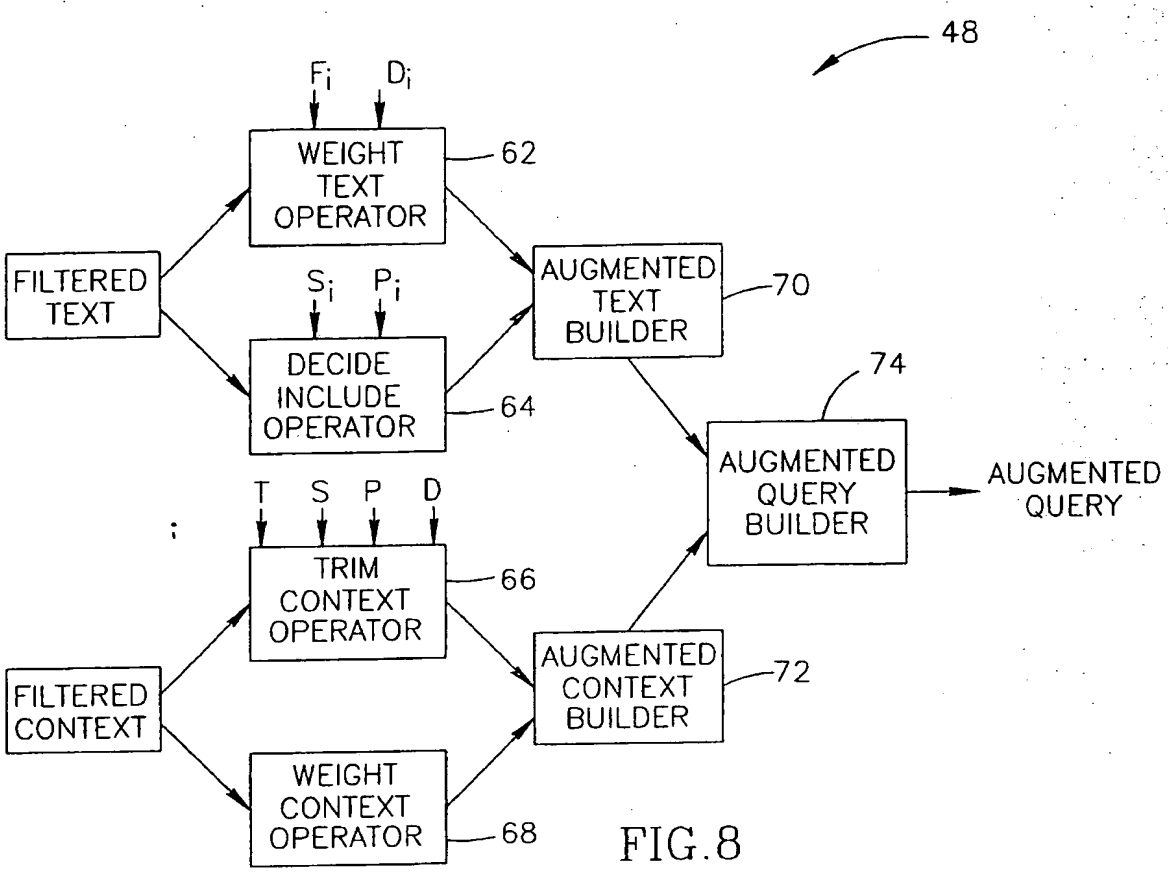

48
F_i
D_i
WEIGHT TEXT OPERATOR
62
FILTERED TEXT
S_i
P_i
DECIDE INCLUDE OPERATOR
64
AUGMENTED TEXT BUILDER
70
74
AUGMENTED QUERY BUILDER
AUGMENTED QUERY
T S P D
TRIM CONTEXT OPERATOR
66
FILTERED CONTEXT
AUGMENTED CONTEXT BUILDER
72
WEIGHT CONTEXT OPERATOR
68

FIG.8

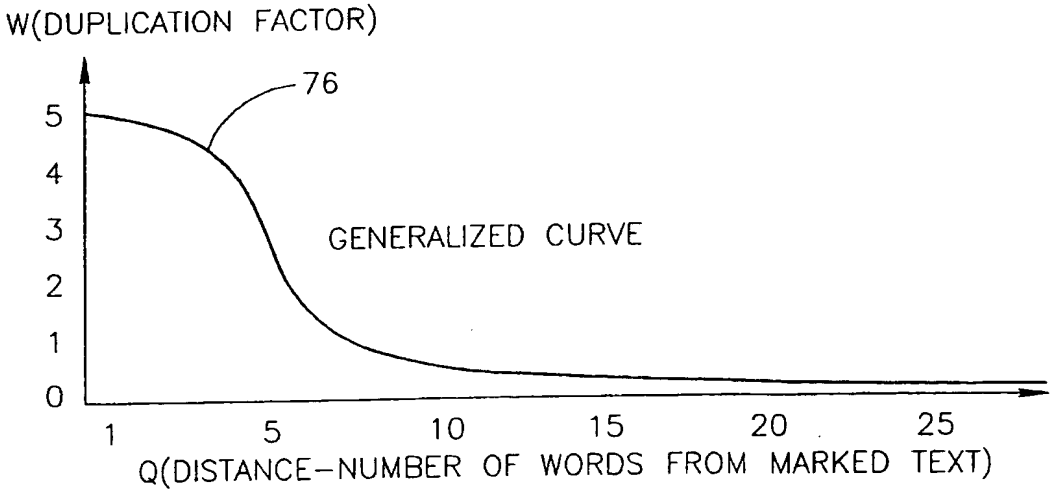

W(DUPLICATION FACTOR)
76
5
4
3
2
1
0
GENERALIZED CURVE
1
5
10
15
20
25
Q(DISTANCE–NUMBER OF WORDS FROM MARKED TEXT)

FIG.9

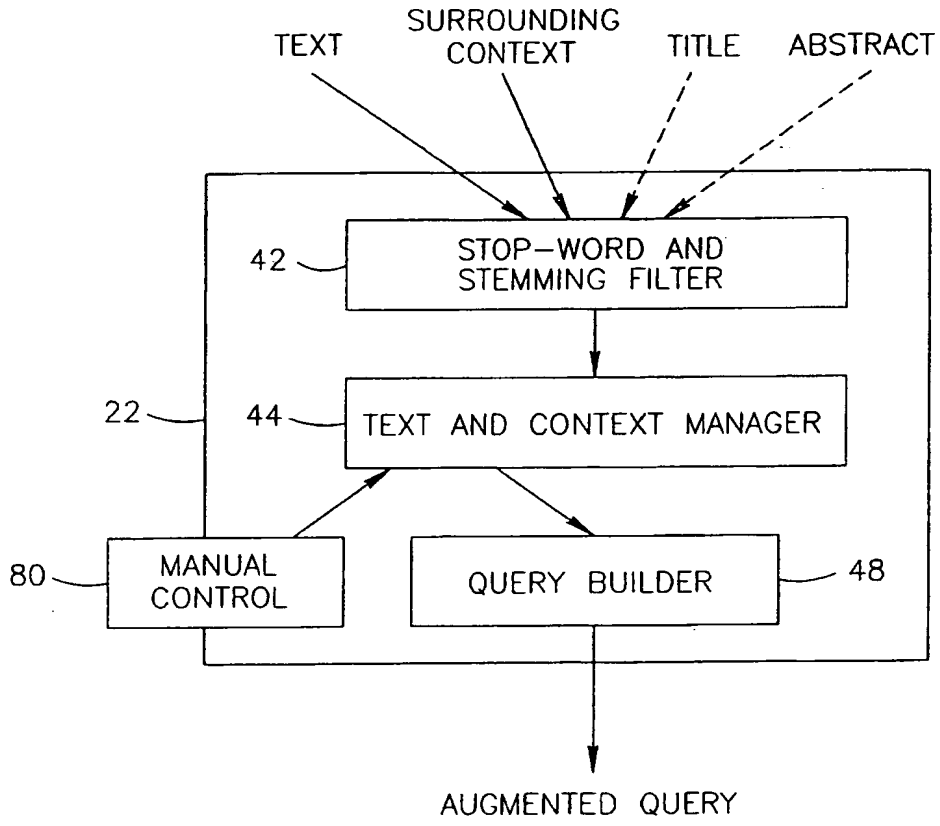

TEXT
SURROUNDING CONTEXT
TITLE
ABSTRACT
42
STOP–WORD AND STEMMING FILTER
22
44
TEXT AND CONTEXT MANAGER
80
MANUAL CONTROL
QUERY BUILDER
48
AUGMENTED QUERY

FIG.10

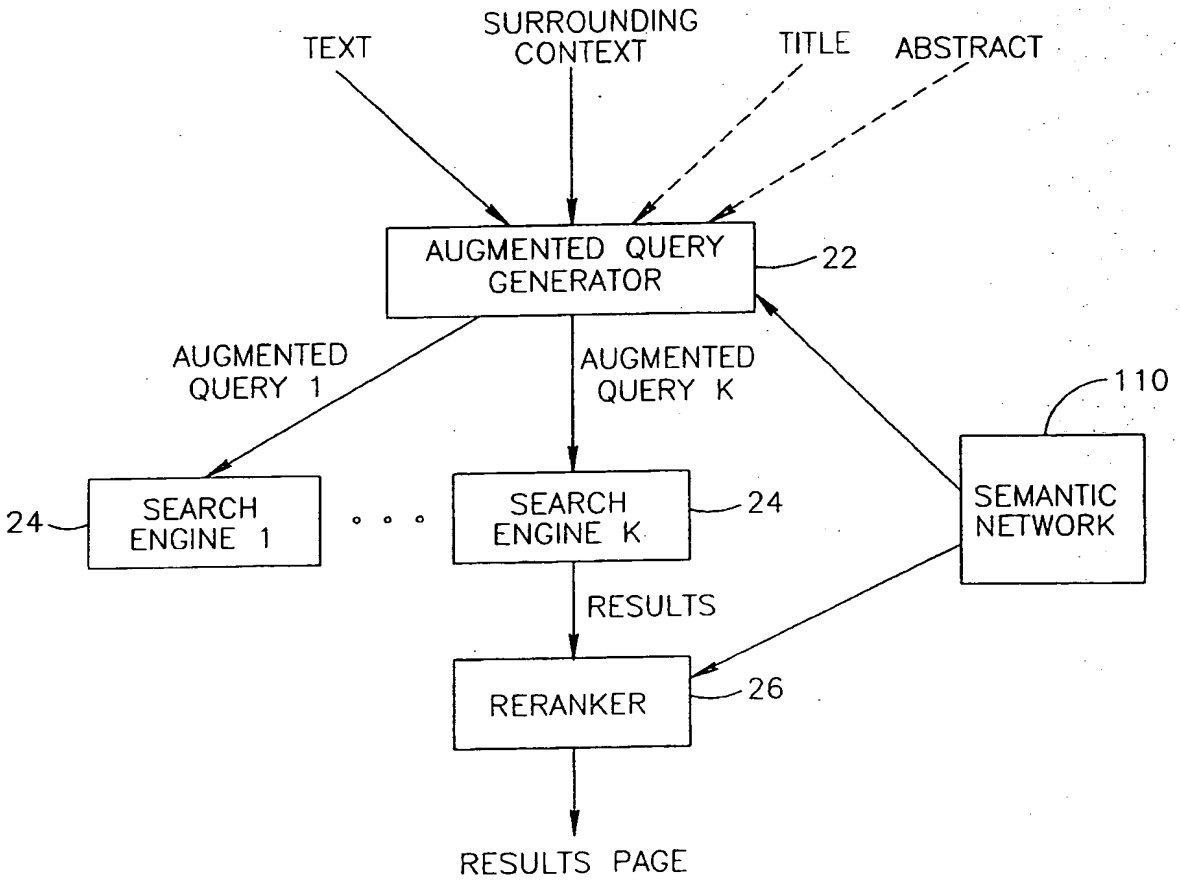

TEXT
SURROUNDING CONTEXT
TITLE
ABSTRACT
AUGMENTED QUERY GENERATOR
22
AUGMENTED QUERY 1
AUGMENTED QUERY K
110
24
SEARCH ENGINE 1
SEARCH ENGINE K
24
SEMANTIC NETWORK
RESULTS
RERANKER
26
RESULTS PAGE

FIG.11

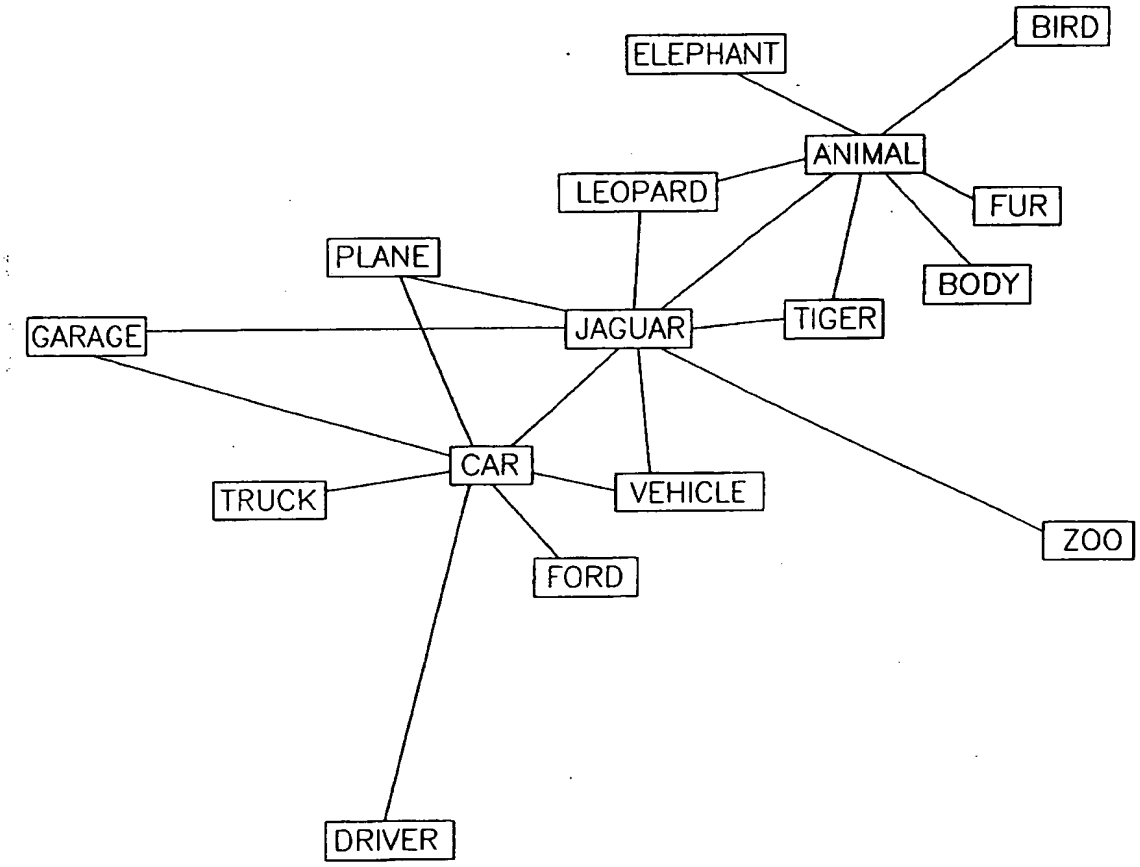

BIRD
ELEPHANT
ANIMAL
LEOPARD
FUR
PLANE
BODY
GARAGE
JAGUAR
TIGER
CAR
TRUCK
VEHICLE
ZOO
FORD
DRIVER

FIG.12

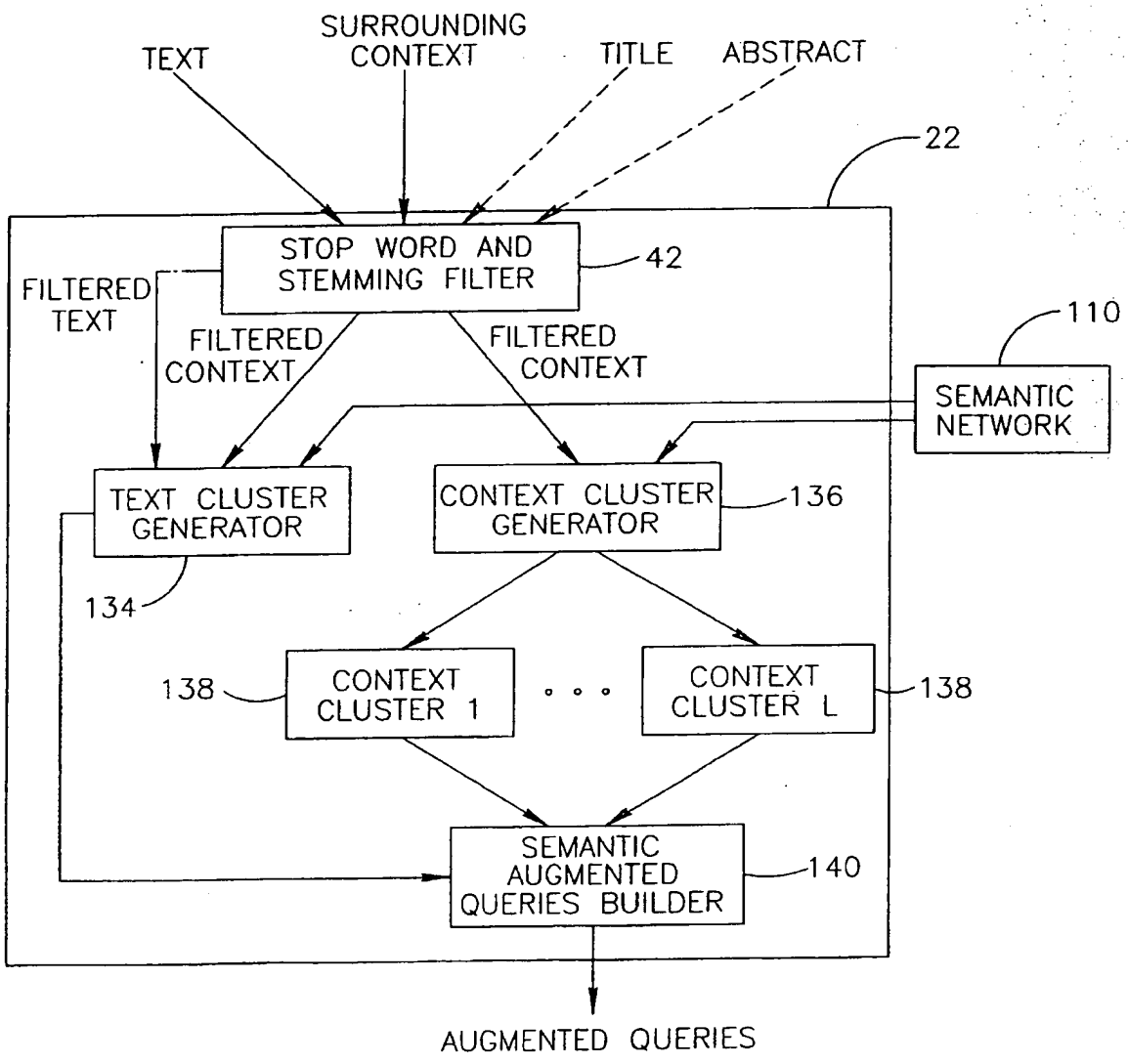

TEXT
SURROUNDING CONTEXT
TITLE
ABSTRACT
22
STOP WORD AND STEMMING FILTER
42
FILTERED TEXT
FILTERED CONTEXT
FILTERED CONTEXT
110
SEMANTIC NETWORK
TEXT CLUSTER GENERATOR
134
CONTEXT CLUSTER GENERATOR
136
138
CONTEXT CLUSTER 1
CONTEXT CLUSTER L
138
SEMANTIC AUGMENTED QUERIES BUILDER
140
AUGMENTED QUERIES

FIG.13

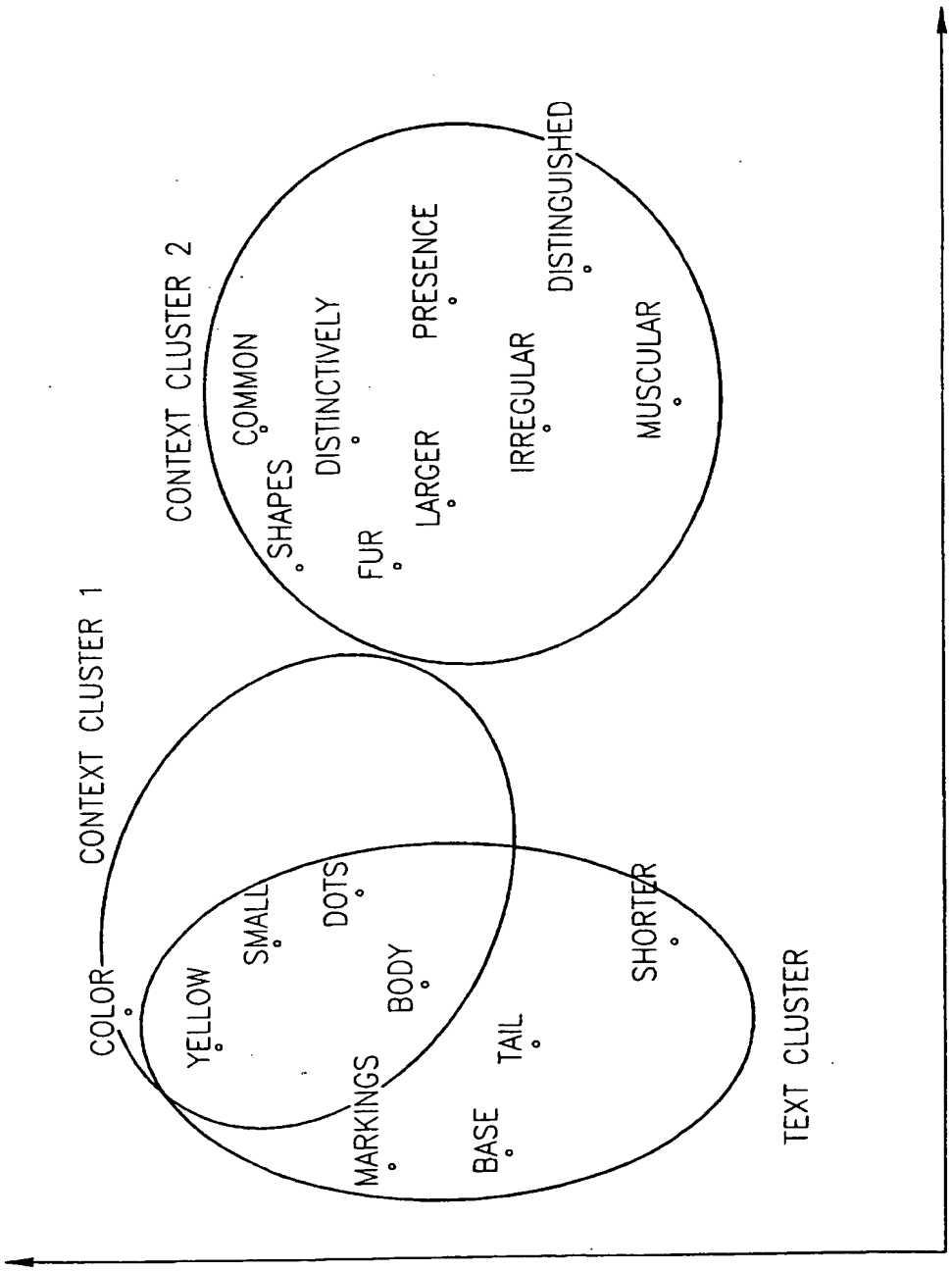

CONTEXT CLUSTER 1
CONTEXT CLUSTER 2
COLOR
YELLOW
SMALL
DOTS
BODY
MARKINGS
TAIL
BASE
SHORTER
TEXT CLUSTER
SHAPES
COMMON
DISTINCTIVELY
FUR
LARGER
PRESENCE
IRREGULAR
DISTINGUISHED
MUSCULAR

FIG.14

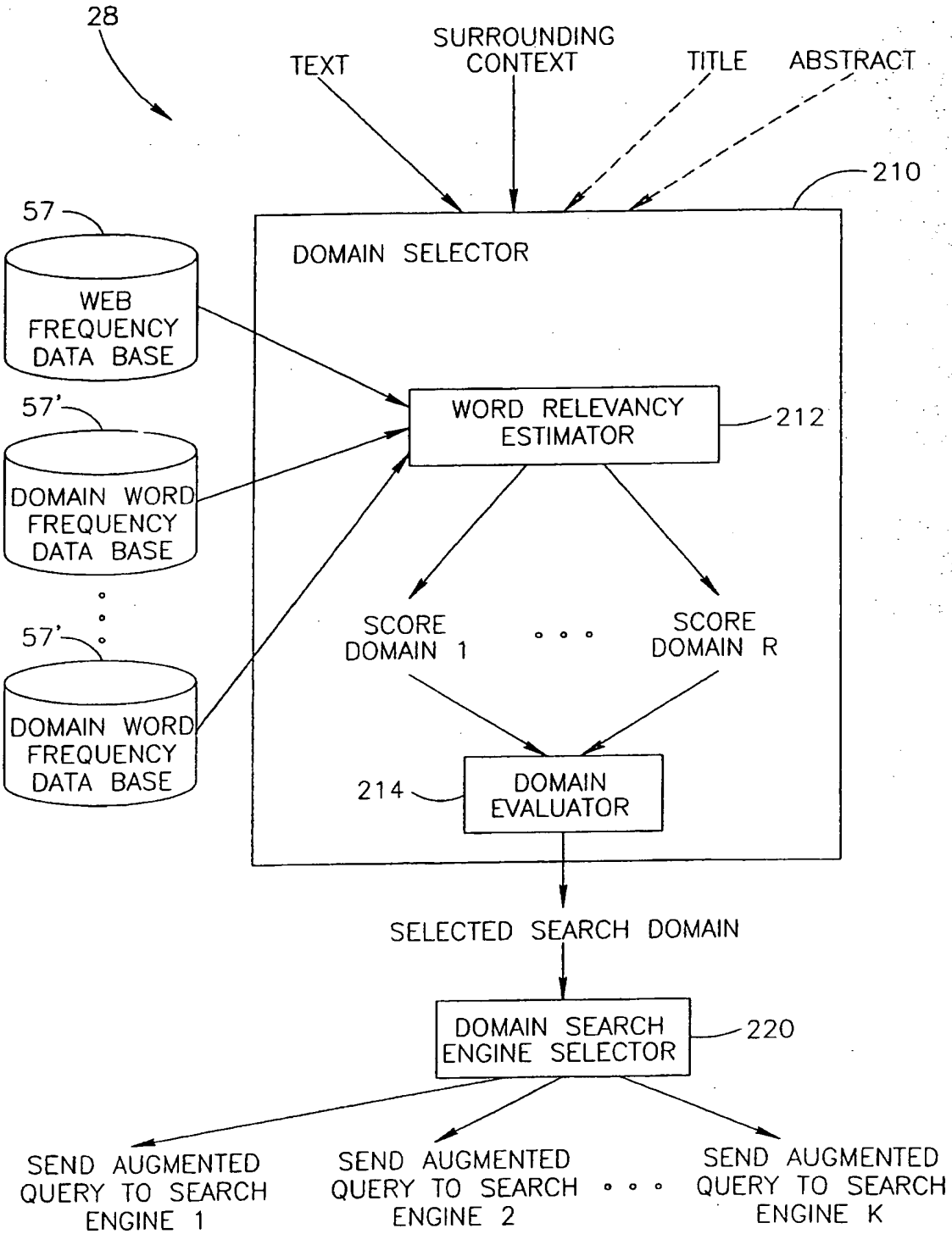

28
TEXT
SURROUNDING CONTEXT
TITLE
ABSTRACT
210
57
WEB FREQUENCY DATA BASE
DOMAIN SELECTOR
57'
DOMAIN WORD FREQUENCY DATA BASE
WORD RELEVANCY ESTIMATOR
212
57'
DOMAIN WORD FREQUENCY DATA BASE
SCORE DOMAIN 1
SCORE DOMAIN R
214
DOMAIN EVALUATOR
SELECTED SEARCH DOMAIN
DOMAIN SEARCH ENGINE SELECTOR
220
SEND AUGMENTED QUERY TO SEARCH ENGINE 1
SEND AUGMENTED QUERY TO SEARCH ENGINE 2
SEND AUGMENTED QUERY TO SEARCH ENGINE K

FIG.15

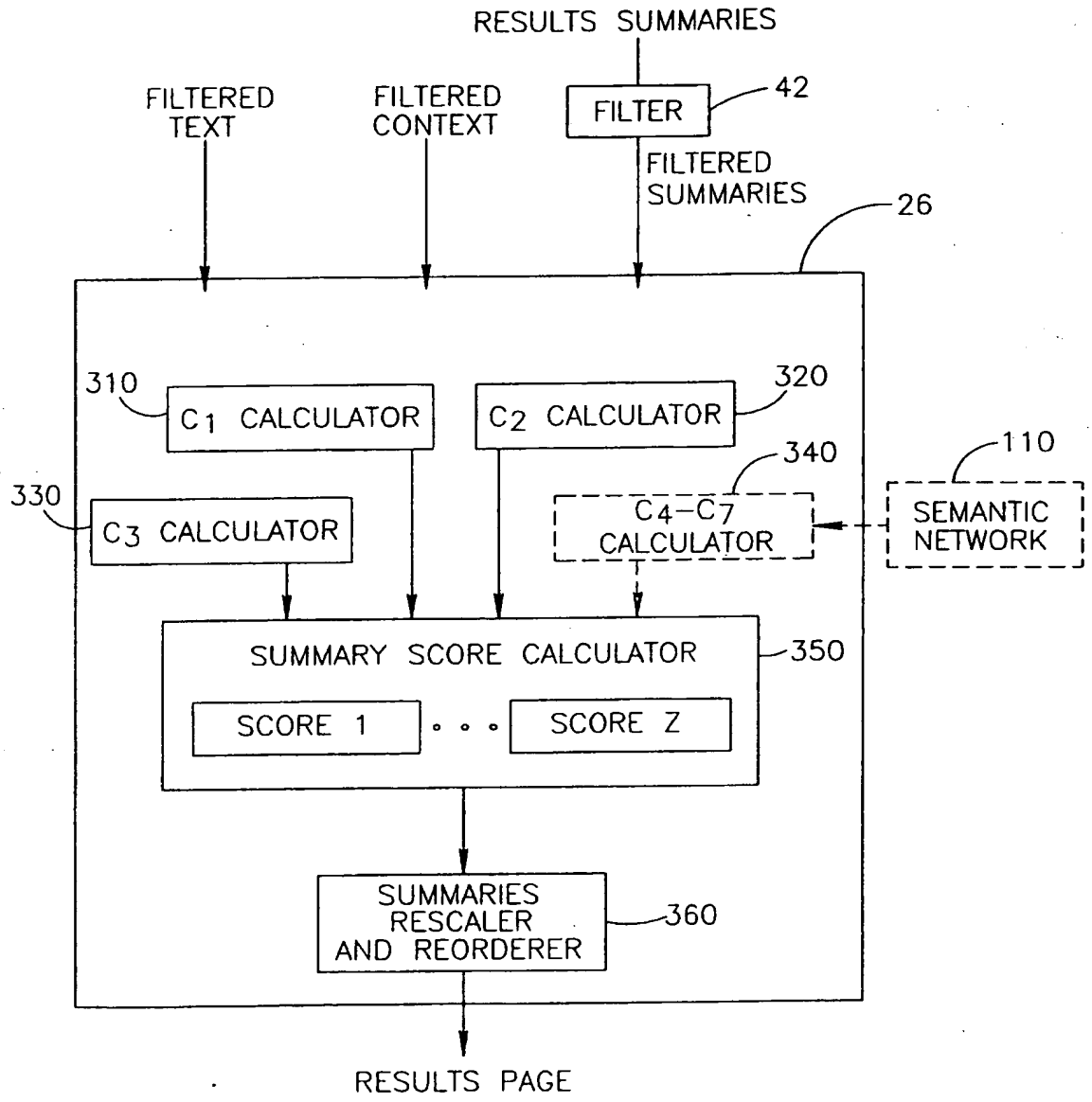

RESULTS SUMMARIES
FILTERED TEXT
FILTERED CONTEXT
FILTER
42
FILTERED SUMMARIES
26
310
C1 CALCULATOR
320
C2 CALCULATOR
340
110
330
C3 CALCULATOR
C4–C7 CALCULATOR
SEMANTIC NETWORK
SUMMARY SCORE CALCULATOR
350
SCORE 1
SCORE Z
SUMMARIES RESCALER AND REORDERER
360
RESULTS PAGE

FIG.16

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IL00/00689

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/00, 17/21, 17/30
US CL : 707/3, 5, 513

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 707/3, 5, 513

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

NONE

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WEST

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,864,846 A (VOORHEES et al.) 26 January 1999, the entire paper is relevant	1-46
Y	US 5,920,856 A (SYEDA-MAHMOOD) 06 July 1999, the entire paper is relevant	1-46
Y	US 5,924,105 A (PUNCH, III et al.) 13 July 1999, the entire paper is relevant	1-46
Y	US 5,933,822 A (BRADEN-HARDER et al.) 03 August 1999, the entire paper is relevant	1-46

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search	Date of mailing of the international search report
02 JANUARY 2001	18 JAN 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer THUY PARDO James R. Matthews Telephone No. (703) 305-9707

Form PCT/ISA/210 (second sheet) (July 1998)★